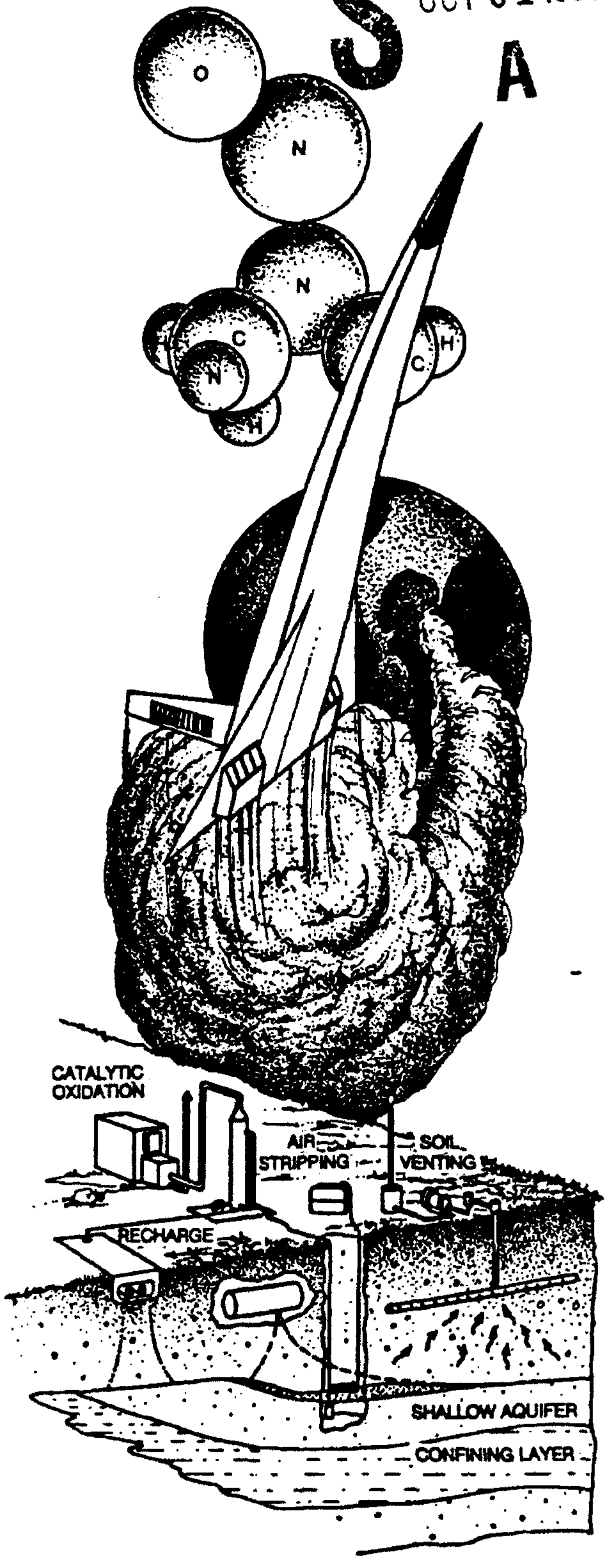
AD-A270 094

DTIC
ELECTE
OCT 01 1993
S A D

(2)

ESL-TR-91-28



HAZARD RESPONSE MODELING UNCERTAINTY (A QUANTITATIVE METHOD) VOL I - USER'S GUIDE FOR SOFTWARE FOR EVALUATING HAZARDOUS GAS DISPERSION MODELS

S. R. HANNA, D. G. STRIMAITIS,
J. C. CHANG

SIGMA RESEARCH CORPORATION
234 LITTLETON ROAD, SUITE 2E
WESTFORD MA 01886

MARCH 1993

FINAL REPORT

APRIL 1989 - APRIL 1991

APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION UNLIMITED

425447

93-22745

ENVIRONICS DIVISION
Air Force Engineering & Services Center
ENGINEERING & SERVICES LABORATORY
Tyndall Air Force Base, Florida 32403

NOTICE

PLEASE DO NOT REQUEST COPIES OF THIS REPORT FROM HQ AFESC/RD (ENGINEERING AND SERVICES LABORATORY). ADDITIONAL COPIES MAY BE PURCHASED FROM:

NATIONAL TECHNICAL INFORMATION SERVICE
5285 PORT ROYAL ROAD
SPRINGFIELD, VIRGINIA 22161

FEDERAL GOVERNMENT AGENCIES AND THEIR CONTRACTORS REGISTERED WITH DEFENSE TECHNICAL INFORMATION CENTER SHOULD DIRECT REQUESTS FOR COPIES OF THIS REPORT TO:

DEFENSE TECHNICAL INFORMATION CENTER
CAMERON STATION
ALEXANDRIA, VIRGINIA 22314

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release Distribution Unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) ESL-TR-91-28
6a. NAME OF PERFORMING ORGANIZATION Sigma Research Corporation 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Air Force Engineering & Service Center
6c. ADDRESS (City, State, and ZIP Code) 234 Littleton Road, Suite 2E Westford, MA 01886	7b. ADDRESS (City, State, and ZIP Code) H.Q. AFESC/RDVS Tyndall Air Force Base, FL 32403
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F08653-89-C-0136
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
Hazard Response Modeling Uncertainty (A Quantitative Method) Unclassified
Volume I: User's Guide for Software for Evaluating Hazardous Gas Dispersion Models

12. PERSONAL AUTHOR(S)
Hanna, S.R., Strimaitis, D.G. and Chang, J.C.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM 89 04 20 TO 91 04	March 1993	

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Toxic Hazards; Dispersion Modeling; Meteorology; Uncertainty in Hazard Response Modeling; Evaluation of Models

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Many microcomputer-based hazard response models for calculating concentrations of hazardous chemicals in the atmosphere are available. The uncertainties associated with these models are not well-known and they have not been adequately evaluated and compared using statistical procedures where confidence limits are determined. The U.S. Air Force needs an objective method for evaluating these models, and this project provides a framework for performing these analyses and estimating the model uncertainties.

This volume of the final report provides a user's guide for the software that has been developed for the quantitative evaluation of the performance of hazardous gas dispersion models. The characteristics and uses of the software are described, the required components of input files are reviewed, and methods of presenting the output files are summarized.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code) / 22c. OFFICE SYMBOL

DD Form 1473, JUN 86 Previous editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE

(The reverse of this page is blank.)

EXECUTIVE SUMMARY

A. OBJECTIVE

The overall objective of this project is to develop and test computer software containing a quantitative method for estimating the uncertainty in PC-based hazardous response models. This software is to be used by planners and engineers in order to evaluate the predictions of hazard response models with field observations and determine the confidence intervals on these predictions. This particular volume (I) is intended to provide the user with guidance for applying the software so that the user can evaluate the models and estimate the associated uncertainties in an objective and systematic fashion.

B. BACKGROUND

The U.S. Air Force and the American Petroleum Institute, among others, have increased emphasis on calculating toxic corridors due to releases of hazardous chemicals into the air. There are dozens of PC-based computer models recently developed in order to calculate these toxic corridors. However, the uncertainties in these models have not been adequately determined, partly due to the lack of availability of a standardized quantitative method that could be applied to these models. Individual model developers generally present a limited evaluation of their own model, and the USEPA has published some partial evaluations, but a comprehensive study has not been completed.

There are two ways to evaluate the performance of a model--statistical (quantitative) and scientific (qualitative). The statistical approach involves computation of various performance measures such as the correlation coefficient, mean bias, fraction within a factor of two, and mean square error. The scientific approach includes the study of the variation of the model residuals (defined as the ratio of the predicted to the observed values) with some primary input parameters such as wind speed and stability class.

DTIC QUALITY INSPECTED 1

Dist
A-1

There are three components of uncertainty in the models: (1) input data errors, (2) concentration fluctuations (stochastic variability), and, (3) errors caused by model physics assumptions. However, these components have not yet been studied in any comprehensive and systematic way.

C. SCOPE

The scope of the overall project has included acquisition and testing of databases and models, development and application of model evaluation software, and assessment of the components of uncertainty.

The current volume (I) is intended to serve as a user's guide to the generic model evaluation software packages, without reference to specific models or issues.

We emphasize in this volume the application aspects of (1) the statistical model evaluation software (BOOT), (2) the scientific model evaluation software (RESIDUAL) using the residual plots, and (3) various routines to investigate model uncertainty due to input data errors and concentration fluctuations. Model uncertainty due to model physics errors is assessed in Volume III.

The blocked bootstrap resampling procedure is used in the BOOT program to estimate the confidence intervals on various model performance measures. Input and output files for the BOOT and RESIDUAL programs are described and presented, and test cases are discussed.

A simple two-dimensional plotting package (SIGPLOT) is also documented in the current volume. The results from both the BOOT and RESIDUAL programs can be easily plotted using the SIGPLOT plotting package.

D. RESULTS

The statistical model evaluation software (BOOT) and the scientific model evaluation software (RESIDUAL) were developed to perform the task of generic model evaluation. As currently implemented, 1000 bootstrap samples are drawn in the BOOT program and used to infer 95 percent confidence intervals for the performance measures. The input files for both programs can be easily

prepared. The input file accepted by the RESIDUAL program is also accepted by the BOOT program, but not vice versa. The mandatory output file generated by the BOOT program is concise and tabular in form. The output file generated by the RESIDUAL program can be plotted using the SIGPLOT plotting package. If required, the BOOT program also generates output files that can be plotted using the SIGPLOT plotting package. Since both the BOOT and RESIDUAL programs are written in Fortran 77, they can be ported to platforms other than personal computers, such as engineering workstations and mainframe computers.

Model sensitivity due to data input errors can be investigated using the Monte Carlo sensitivity analyses. Sampling routines are available to help the user randomly select a number from the following five probability density functions: uniform, exponential, normal, log-normal, and clipped normal. However, the user has to develop his own main program to implement the Monte Carlo sensitivity analyses of a model and to use these sampling routines.

Recent research studies show that concentration fluctuations (the stochastic component of uncertainty) are a function of many atmospheric variables. In order to assure that the algorithm for estimating concentration fluctuations will be robust, only empirical formulas are suggested in the current volume. Concentration fluctuations are assumed to depend only on crosswind and vertical distances from the centerline, dispersion coefficients, and concentration averaging times. Contributions to stochastic uncertainty from other components will be included in the future.

E. CONCLUSIONS AND RECOMMENDATIONS

Generic statistical model evaluation (BOOT) and scientific model evaluation (RESIDUAL) software packages were developed and described. They can be used to gauge the performance of any type of model. Their usage is not limited to hazard response models or air quality models. Because one of the shortcomings of any statistical model evaluation is that the performance measures are sometimes overly influenced by outliers, it is recommended that the data be scaled or transformed in order to minimize the influence of outliers.

Implementation of the Monte Carlo sensitivity analyses depends on the particular application at hand; therefore, a "turnkey" software package is not

available. Nevertheless, recommended procedures to implement the Monte Carlo sensitivity analyses have been outlined and the sampling routines are available to the user.

The subject of concentration fluctuations is a complex matter and still not fully understood. We choose to use only the most practical formulas, as described before, in order to assure the robustness of the algorithm for estimating concentration fluctuations. More research on this subject matter is clearly needed.

PREFACE

This report was prepared by Sigma Research Corporation, 234 Littleton Road, Suite 2E, Westford, Massachusetts 01886, under the Small Business Innovative Research (SBIR) Phase II program, Contract Number F08635-89-C-0136, for the Air Force Engineering and Service Center, Engineering and Services Laboratory (AFESC/RDVS), Tyndall Air Force Base, Florida 32403. The project has been cosponsored by the American Petroleum Institute, 1220 L Street Northwest, Washington DC 20005 under Project Number AQ-7-305-8-9.

This report summarizes work done between 20 April 1989 and 20 April 1991. AFESC/RDVS project office was Captain Michael Moss and API project officer was Mr. Howard Feldman. This report has three volumes. Volume I is entitled User's Guide for Software for Evaluating Hazardous Gas Dispersion Models. Volume II is entitled Evaluation of Commonly-Used Hazardous Gas Dispersion Models, and Volume III is entitled Components of Uncertainty in Hazardous Gas Dispersion Models.

Because this is an SBIR report, it is being published in the same format in which it was submitted.

This report has been reviewed by the Public Affairs (PA) Office and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

Mike Moss
MICHAEL T. MOSS, Maj, USAF
Chief, Environmental Compliance R&D

Neil J. Lamb.
NEIL J. LAMB, Col, USAF, BSC
Chief, Environics Division

Frank P. Gallagher
FRANK P. GALLAGHER III, Colonel, USAF
Director, Air Force Civil Engineering Laboratory

(The reverse of this page is blank.)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (CONCLUDED)

LIST OF TABLES

SECTION I

INTRODUCTION

A. OBJECTIVES AND SCOPE

There are no standard objective quantitative means of evaluating microcomputer-based hazard response models. Dozens of such models have been recently proposed, and many of them include up-to-date algorithms on important scientific phenomena such as two-phase jets, evaporative emissions, dense-gas slumping, and transition to non-buoyant dispersion. The Air Force has sponsored the development of some of these models, such as ADAM, AFTOX, CHARM, DEGADIS, SLAB, and OB/DG. A few data sets exist for testing these models, but the models have not been tested or compared with a comprehensive set of these data on the basis of standard statistical significance tests. Limited testing has been done using field data sets from older experiments such as the Prairie Grass field studies and more recent experiments such as the Eagle and Desert Tortoise field studies.

The U.S. Air Force, among others, has increased emphasis on calculating "toxic corridors" caused by potential release of hazardous chemicals. The Ocean Breeze/Dry Gulch (OB/DG) model was originally used for calculating these corridors, and does contain an estimate of model uncertainty. However, the OB/DG model does not account for the important scientific phenomena mentioned above. The new models mentioned above are more advanced scientifically, but do not include model uncertainty. The objective of the current research project has been to fully develop these quantitative procedures, to better estimate the components of the uncertainty (data input errors, stochastic uncertainties, and model physics errors), and to test the procedures using a wide spectrum of field and laboratory experiments.

The results of the research project are presented in three volumes:

I. User's Guide for Software for Evaluating Hazardous Gas Dispersion Models

II. Evaluation of Commonly-Used Hazardous Gas Dispersion Models

III. Components of Uncertainty in Hazardous Gas Dispersion Models

The current volume (I) is intended to serve as a user's guide to the generic model evaluation software, without reference to specific models or issues. Input and output files are presented, and test cases are discussed.

B. OVERVIEW OF RELATED EPA PROCEDURES

The U.S. EPA has been developing and applying computerized model evaluation procedures for over 10 years. Recently they published a user's guide for their procedures (Reference 1). Their User's Guide documents a computerized system for comparing the performance of two or more air quality simulation models. The methodology is based on procedures that have been recommended by EPA and described in a companion document entitled Procedures for Determining the Best Performing Model, dated August 1988. A more technical discussion of the statistical techniques used in this procedure is given by Cox and Tikvart (Reference 2).

To use the EPA system, a database must be available that contains ambient measurements, meteorological data, and concentrations that have been predicted using two or more simulation models. Emphasis is on databases containing 1 year of hourly data.

Model accuracy is defined in terms of the difference between the measured and model predicted concentrations, both for individual station/meteorological combinations and for maximum concentrations over the network of stations during the modeled time period. The bootstrap resampling technique is used to establish confidence bounds on various measures of model performance including the composite performance of each model and the difference in performance for the two models being considered.

The EPA system operates within a TSO environment on EPA's IBM mainframe computer located in the Research Triangle Park, NC. All computations are coded in SAS while Command Lists have been used to create a menu of panels to assist the user in executing the system. The interactive portion of the system invokes a sequence of screen panels which prompt users for information regarding data files, input parameters and other data needed to calculate and process model performance statistics.

The current USAF/API model evaluation system is intended to provide several improvements over the EPA system, with emphasis on hazard response models that operate on PC's.

SECTION II

OVERVIEW OF FRAMEWORK OF SOFTWARE

The current volume (I) presents the user's guide for the generic model evaluation software. The goal is to establish well-defined procedures to better evaluate the performance and to better estimate the uncertainty of the models.

There are two ways to evaluate the performance of a model--statistical (quantitative) and scientific (qualitative). The statistical evaluation of a model involves the calculation of performance measures such as the correlation coefficient. However, the quantitative performance measures sometimes suffer from the following two shortcomings: (1) they do not provide enough insight into why a model performs as it does, and (2) some of the quantitative measures are overly influenced by outliers. Consequently, our software includes algorithms for assessing both the statistical and scientific performance of models.

The statistical evaluation of a model includes determination of the fractional mean bias, the normalized mean square error, the correlation coefficient, and the fraction of the predictions that are within a factor of two of the observations. Confidence intervals on these quantities and on the differences between models are estimated using the bootstrap resampling procedure.

The scientific evaluation of a model involves the investigation of the variation of the model residuals, defined as the ratio of the predicted to the observed concentrations, with primary parameters such as wind speed, stability and downwind distance. Therefore, for example, if the residuals of a model exhibit a systematic bias only when the atmosphere is stable, it is possible that the dispersion algorithms used by the model during stable conditions require modification.

Model uncertainty due to data input error can be investigated using Monte Carlo sensitivity analyses, as discussed in Section IX.

Included in the subsequent sections of this volume are the user's guides for various programs to evaluate model performance and estimate model uncertainty. As stated before, our goal is to make the programs as generic as possible. There are no references to any specific implementations. As a result, some of the programs are in the form of subroutines. Depending on the application, the user has to develop the main program that will use some of these subroutines, such as the Monte Carlo sensitivity analysis. An example would be the MDAMC software package (described in Volume II of this report) that uses the RAN1 subroutine (described in Section IX of the current volume) to investigate model uncertainty due to data input error. Recommendations or suggestions are given below to help the user identify important issues or potential problems during the actual implementation of this software.

A brief description of the software is given below:

- Quantitative statistical model evaluation software (BOOT):

  The BOOT program calculates various quantitative statistical performance measures for the models. It also uses the blocked bootstrap resampling procedure (Reference 3, Reference 4) to estimate the confidence limits on these performance measures. The BOOT program is a complete, self-contained program that accepts a simple input.

- Residual analysis program (RESIDUAL):

  The RESIDUAL program analyzes the distribution of the model residuals as a function of the primary parameters. The RESIDUAL program is a complete, self-contained program that accepts a simple input file. The same input file can be used by both the BOOT and RESIDUAL programs.

- Sampling routines for five probability distribution functions (pdf):

  Model sensitivity due to data input errors can be investigated using the Monte Carlo sensitivity analyses. Sampling routines are available to help the user to randomly select a number from the following five pdf's: uniform, exponential, Gaussian, log-normal, and clipped normal. The user has to develop his own main program to implement the Monte Carlo sensitivity analyses of a model and to use these sampling routines.

- Stochastic uncertainty estimation program (ESTSIG):

  Practical formulas for estimating stochastic uncertainty (that is, concentration fluctuations) are implemented in this subroutine. It was coded with future expansion in mind so that more theoretical equations can be included later, after they have been validated. The user has to develop his own main program to use this subroutine.

- Two-dimensional plotting package (SIGPLOT):

  The SIGPLOT plotting package is a versatile tool for producing many kinds of two-dimensional plots. The results generated by the BOOT and RESIDUAL programs can be readily plotted using the SIGPLOT plotting package.

SECTION III

DEFINITION OF GOALS OF A SPECIFIC STUDY

The software described in this User's Guide is generic in that it can be applied, in whole or in part, to any model evaluation study. The software is equally valid for many types of models, including air quality models, weather forecast models, economic models, health risk models, and so on. Given an input file consisting of a series of two to nine columns of numbers, the software will analyze the relations between those columns of numbers, no matter what the numbers mean or where they were obtained.

In view of the ability of the software to produce seemingly limitless sets of statistics, tables, and figures, it is important to clearly define the goals of a specific study. The proper hypotheses or questions must first be asked. By this means, the results can be more clearly interpreted. An example of a question that might be asked would be:

> Is there a significant difference between the predictions of models A and B, when applied to experiment C, where the data represent 5 minute averages of maximum concentration observed anywhere on monitoring arcs at three downwind positions?

Another question might be:

> What is the sensitivity of the predictions of Model D in a given source-receptor scenario to variations in input parameters E and F?

or:

> Which of the models G, H, I, J, K produces the least variation of model residuals with wind speed for experiments L, M, and N?

Different subroutines in the software would be used to answer these three questions. Furthermore, depending on the goals, different models or data sets might be considered, as discussed in the next two sections.

SECTION IV

MODEL ACQUISITION

Perhaps a scientist or engineer is lucky in the sense that he already possesses the models to be evaluated, is familiar with their use, and these models are relatively unchanging (i.e., they are not in a constant state of modification). However, in most model evaluation exercises (including the one described in Volume II), it is necessary to select and acquire several models for evaluation.

Criteria for model selection should be defined, and could include the following items:

- Cost of model
- Type of computer required
- Speed of model
- Applicability of model to the scenario of interest
- Availability of user's guide and/or technical description
- Can input data needs be fulfilled?
- Are the output data in the form required?
- Is the source code available?
- Is the model of interest to the sponsor of the study?
- Is a stable (i.e., unchanging) version of the model available?
- Are the developers available for guidance?

Testing of the model should take place after it is acquired. This should include both the test case that comes with the code and one or more scenarios more closely related to the scenarios to be studied. The input and output needs should be reviewed at this stage.

Scientific review of the technical description of the model should occur at this stage, if possible, to determine if the scientific algorithms in the model are correct. This is an optional step, but can be crucial for some new models or studies. For example, it may be found that the model assumes an area source rather than a point source, which may result in a decision that the model is inapplicable to experiments involving point sources. Or, it may be found that certain coefficients or formulas in the code are inconsistent with those in the technical documentation.

SECTION V

DATA ACQUISITION

While it is possible to use the model evaluation software to compare the predictions of two or more models, with no concern with observations, in most cases, the predictions of models are compared with observations. In this case, the scientist or engineer must acquire one or more sets of experimental data. He may already have these data on hand, or he may need to obtain them from other persons.

The data to be acquired are closely connected with the goals defined for the study, which describe the source scenarios, downwind distances, elevations, etc., of interest. Of course the data should also satisfy QA/QC requirements, be of reasonable cost, and be available in a format that permits ease of use (for example, magnetic tapes or floppy disks). A technical report should be available that thoroughly describes the experiment.

Once the data sets are acquired, they should be checked to be sure nothing is missing. The technical report should be consulted to identify and remove questionable data and decide upon methods for blocking (that is, dividing the data into similar groups according to criteria such as downwind distance, source type, etc.).

The data should be placed in a "Modelers' Data Archive" (MDA) sufficient for running all of the models and conducting the evaluation. If certain models require the input of parameters not in the original data archive, such as molecular weights, ambient air densities, or latent heats of vaporization, these parameters must be somehow determined or calculated and inserted in the MDA. The MDA also contains the observed concentrations in a format to be used for model evaluation.

The model predictions will sometimes already be available and there is no need to construct an MDA. If this occurs, then the only data needed are the observed concentrations, and the locations and averaging times of these observations. Concurrent values of wind speed, stability, source strength, etc., may be included for the residual analysis. Expected uncertainties of each of these variables (see Section IX for more detail) are needed to apply the software used for carrying out the Monte Carlo sensitivity analysis.

SECTION VI

DEFINITION OF NEEDS FOR PREPROCESSING AND POSTPROCESSING

If all models do not start with the same input and end at the same output, preprocessing and postprocessing algorithms may need to be developed. For example, the predicted width of the plume may be of interest, but only two of six models being evaluated may include the width in their output. It is then necessary to develop methods (for example, code inserted in the main program or applied as a postprocessor) for producing these widths. Some models may predict the source emission rate based on the physical properties of a storage tank, while other models may assume that the source emission rate is given. In this example, preprocessing software must be written so that the models begin on equal footing.

If a model does not exactly match the source scenario (for example, the model does not treat aerosols while the experiment deals with a two-phase release of ammonia), it may be necessary to use guidance in the literature to modify the input parameters so that they best represent the source type. This could be done, for example, by modification of the assumed value for the initial plume density.

If too much pre- and postprocessing is needed, the model evaluation exercise is pointless. Furthermore, some users may not be familiar with the physical and chemical relations that must be known. If the models and data are too inconsistent, the model evaluation software should not be applied.

SECTION VII

QUANTITATIVE STATISTICAL MODEL EVALUATION SOFTWARE

The model evaluation software package, BOOT, described in this section is based on recommendations by Hanna (Reference 3), who has applied an earlier version of the software to several air quality modeling scenarios.

The BOOT program calculates the model performance measures known as the fractional bias (FB), geometric mean bias (MG), normalized mean square error (NMSE), geometric mean variance (VG), correlation coefficient (R), fractional variance (FS), and fraction within a factor of two (FAC2), which are defined below:

$$FB = \frac{\overline{C_o} - \overline{C_p}}{0.5(\overline{C_o} + \overline{C_p})} \quad (1)$$

$$MG = \exp(\overline{\ln C_o} - \overline{\ln C_p}) \quad (2)$$

$$NMSE = \frac{\overline{(C_o - C_p)^2}}{\overline{C_o C_p}} \quad (3)$$

$$VG = \exp\left[\overline{\left(\ln C_o - \ln C_p\right)^2}\right] \quad (4)$$

$$R = \frac{\overline{(C_o - \overline{C_o})(C_p - \overline{C_p})}}{\sigma_{C_p}\sigma_{C_o}} \quad (5)$$

$$FS = \frac{\sigma_{C_o} - \sigma_{C_p}}{0.5(\sigma_{C_o} + \sigma_{C_p})} \quad (6)$$

FAC2 = fraction of data which $0.5 \le C_p/C_o \le 2$. (7)

where C_o is the observation, and C_p is the model prediction. The software package, written in FORTRAN, uses the blocked bootstrap resampling method (Reference 3, Reference 4) to estimate the confidence limits on these performance measures. The user instructs the BOOT program how to partition the data points into many blocks (if necessary) in the input file (described later). Because the bootstrap samples are drawn from the blocked groups of data, the additional variance due to the mean bias between data blocks is not included. As currently implemented, 1000 bootstrap samples are taken and used to infer the confidence limits. The output generated by the BOOT program is highly compact and tabular in form. The program also generates files containing information on FB or MG (including confidence limits), and NMSE or VG, that can be plotted using the SIGPLOT plotting package (see Appendix A). The choice of FB versus MG, and NMSE versus VG depends on the program option (described later) selected by the user.

The BOOT program requires up to two input files and generates up to three output files. Only the random number input file has an assumed name, RANDS; the user is prompted for the names of the other files during the execution of the program. A description of each file is given below:

- Input Files:

  The mandatory input data file, supplied by the user, contains multiple columns of data, representing concurrent values of the observations and the model predictions. Table 1 describes the format of the input file. Figure 1 shows an example of the input file. Note that the BOOT program has no provisions for correcting for missing data. It is the responsibility of the user to assure that real data exist at each position in the file.

  The optional random number file, RANDS, is an input file containing a series of random numbers. This file is the only one whose name is preassigned in the BOOT program. It will be opened and consulted only if the user decides to use the bootstrap resampling procedure to estimate the confidence limits of the performance measures.

TABLE 1. FORMAT FOR THE MANDATORY INPUT DATA FILE OF THE BOOT PROGRAM. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL.

LINE NO.	FORMAT	DESCRIPTION
1	FF/I	There are three integer constants in this line, representing the total number of observations (NN, < 251), the total number of models (MM, < 16), including the observed (or some baseline model) as one, and the total number of blocks (KK < 17) of data. The limits on NN, MM, and KK are assigned in the program using the PARAMETER statements, and can be easily changed. Let KK = 1 if no blocking is desired.
2	FF/I	There are KK integer constants in this line, representing the number of pieces in each block. Note that the sum of all these integers must equal NN.
3	FF/I	There are MM character constants in this line; each one can be at most eight characters long, containing the name of each of the models. All character constants must be enclosed in apostrophes.
4	FF/I	There are KK character constants in this line; each one can be at most 20 characters long, containing the name of each of the blocks. All character constants must be enclosed in apostrophes.
Next NN lines	FF/R	There are MM real numbers in each line, with the first number representing the observed value (or the prediction based on some base line model) and the following MM-1 numbers representing the prediction from each of the remaining models.

```
  79     4  2
  39 40
'OBS.' 'MODEL-A' 'MODEL-B' 'MODEL-C'
'Urban data set' 'Rural data set'
 616.0  708.7  594.7  516.5
 604.1  689.2  585.8  496.7
 868.0  674.8  580.3  516.8
 498.6  668.8  652.1  548.3
 393.1  560.2  704.7  581.9
 409.0  740.9  570.1  621.4
 640.2  249.6  510.1  553.5
 265.3  259.6  463.4  446.0
 192.7   91.6  131.0  485.0
1149.1 1217.5 1116.1  520.6
 972.8 1275.8 1175.1  536.9
1137.5 1225.7 1081.7  617.4
 669.5 1052.8  905.1  637.3
 595.5  862.0  862.0  664.1
 741.2  589.5  767.0  665.3
 612.6  602.4  728.2  672.4
 312.0  398.9  657.5  659.5
 400.2  340.2  412.3  586.0
 264.7  612.1  774.2  705.9
 290.0  428.4  757.3  708.8
 459.5  355.0  512.3  602.4
 444.0  216.0  441.4  681.1
 175.1  216.6  456.1  825.4
 102.3  126.1  255.6  522.9
 128.8   16.5    0.5  834.9
 200.2  301.9  208.9  728.0
 358.3  481.8  354.0  742.4
 611.1 1010.2  987.1  679.0
 499.3  752.5  921.6  725.7
 537.8  724.0  826.8  675.9
 220.0  523.3  908.2  640.8
 479.2  357.5  788.6  544.7
 133.2  195.3  383.1  738.5
  98.2  167.3  213.5 1064.9
  92.5  104.6  142.2  741.2
  21.0  127.4  176.3  805.2
 353.0  307.8  167.1  576.9
 358.0  280.9  188.4  225.3
 233.3  355.3  234.9  719.1
```

Figure 1. An Example of the Input Data File for the BOOT Program. Refer to Table 1 for the Format of the File. In Brief, the Example Shows that there are 79 Data Points from 4 Models, under the Names "OBS.", "MODEL-A", "MODEL-B", and "MODEL-C", Respectively. The 79 Data Points are Divided into two Blocks; the first Block, "Urban Data Set", includes the first 39 Points; the second Block, "Rural Data Set", includes the next 40 Points. The Main Section of the Data File Lists the Concurrent Values from the 4 Models.

198.3	12.7	184.0	745.2
507.2	0.0	126.3	664.9
313.7	0.0	0.0	667.1
165.1	0.0	0.0	703.9
295.6	329.9	454.6	695.3
527.7	308.0	295.9	775.0
454.1	301.0	1.0	995.6
240.3	417.5	361.1	933.8
590.8	579.3	144.2	666.5
638.3	756.6	608.9	400.1
949.8	1004.2	805.4	528.9
886.8	855.6	706.2	517.4
635.5	761.0	670.9	596.6
359.3	412.6	232.5	937.6
484.7	360.7	226.8	979.0
529.7	332.0	202.5	980.0
585.8	291.4	186.1	1100.1
367.7	368.0	260.2	1005.6
324.7	270.9	72.7	1058.6
489.0	274.6	208.5	942.2
570.8	337.1	218.0	646.5
419.7	254.4	206.1	344.0
532.8	414.2	197.9	477.0
425.2	365.7	198.7	469.5
467.5	411.5	228.5	455.3
362.2	306.4	147.6	405.2
429.2	287.4	139.2	450.6
446.0	338.1	169.5	461.2
192.9	253.8	145.6	460.7
630.3	322.5	257.2	460.5
364.9	326.7	251.1	510.6
111.4	196.4	248.5	0.0
89.8	146.5	254.9	0.0
82.5	248.0	160.9	0.0
296.5	253.2	193.2	0.0
215.4	299.7	165.0	0.0
454.5	274.2	154.0	0.0
384.7	324.6	163.2	0.0
253.2	488.3	175.6	0.0
289.5	304.1	193.1	0.0

Figure 1. An Example of the Input Data File for the BOOT Program. Refer to Table 1 for the Format of the File. In Brief, the Example Shows that there are 79 Data Points from 4 Models, under the Names "OBS.", "MODEL-A", "MODEL-B", and "MODEL-C", Respectively. The 79 Data Points are Divided into two Blocks; the first Block, "Urban Data Set", includes the first 39 Points; the second Block, "Rural Data Set", includes the next 40 Points. The Main Section of the Data File Lists the Concurrent Values from the 4 Models (Concluded).

- Output Files:

  The mandatory output file presents results of the BOOT program in a highly compact and tabular fashion. It contains: (1) the calculated values of the performance measures for each of the models, (2) detailed information about the confidence limits, if requested by the user, and (3) tables summarizing the quantitative results of the analysis of confidence limits. An example of this output file is listed in Figure 2.

  The optional FB (or MG) vs. NMSE (or VG) file contains the fractional bias FB (or MG) (with its 95 percent confidence limits) and the normalized mean square error NMSE (or VG) for each of the models. The information stored in this file can be plotted using the SIGPLOT plotting package. This file will be created only if the user decides to use the bootstrap resampling procedure. Figure 3 shows an example of this file plotted using SIGPLOT.

  The optional D(FB) (or D(MG)) vs. D(NMSE) or (D(VG)) file contains the differences in the fractional bias D(FB) (or (D(MG)) (with its 95 percent confidence limits) and the normalized mean square error D(NMSE) (or D(VG)) between pairs of the models. The information stored in this file can be plotted using the SIGPLOT plotting package. This file will be created only if the user decides to use the bootstrap resampling procedure. Figure 4 shows an example of this file plotted using SIGPLOT.

During the execution of the BOOT program, the following questions will be asked:

- Name of the input file:

  The user specifies the name of the mandatory input data file here. There is no default answer. For example, if the input data are contained in a file called "TEST.DAT" residing in the current default director, then the user should type "TEST.DAT" here. If the input data are contained in a file called "WIDGET.INP" residing in the directory "C:\USR", then the user should type "C:\USR\WIDGET.INP" here.

```
OUTPUT OF THE BOOT PROGRAM, LEVEL 910514

No. of observations =  79
No. of models =         4
No. of blocks =         2
No. of pieces in each block
      39  40

Out of the following menus,
(1) straight Co and Cp comparison
(2) consider Co/Co and Cp/Co
(3) consider Co/Co and Co/Cp
(4) consider ln(Co) and ln(Cp)
 1 was selected by the user

All observations,                  (N=    79)
model        mean       sigma        bias       nmse    cor     fa2      fb      fs
OBS.       426.58      235.39        0.00       0.00  1.000   1.000   0.000   0.000
MODEL-A    426.04      286.73        0.54       0.18  0.784   0.835   0.001  -0.197
MODEL-B    402.67      297.02       23.91       0.34  0.612   0.570   0.058  -0.232
MODEL-C    580.37      270.14     -153.79       0.58  0.065   0.544  -0.305  -0.137

Block    1: Urban data set         (N=    39)
model        mean       sigma        bias       nmse    cor     fa2      fb      fs
OBS.       439.41      273.79        0.00       0.00  1.000   1.000   0.000   0.000
MODEL-A    509.45      329.36      -70.05       0.16  0.847   0.821  -0.148  -0.184
MODEL-B    569.11      304.22     -129.70       0.24  0.747   0.718  -0.257  -0.105
MODEL-C    636.27      134.77     -196.86       0.57 -0.384   0.590  -0.366   0.681

Block    2: Rural data set         (N=    40)
model        mean       sigma        bias       nmse    cor     fa2      fb      fs
OBS.       414.08      189.82        0.00       0.00  1.000   1.000   0.000   0.000
MODEL-A    344.72      207.89       69.36       0.20  0.709   0.850   0.183  -0.091
MODEL-B    240.39      175.10      173.69       0.58  0.592   0.425   0.531   0.081
MODEL-C    525.86      346.98     -111.78       0.59  0.312   0.500  -0.238  -0.586
```

Figure 2. An Example of the Output File for the BOOT Program, Where the Model Performance Measures are based on the C_o and C_p Values Listed in Figure 1. If the User Chooses to Work with the Logarithms of C_o and C_p (Accomplished as a Program Option), then FB will be Replaced with MG, and NMSE will be Replaced with VG.

Note: The seductive 95% confidence limits are based on the 2.5% and 97.5% points on the cumulative distribution function.
The robust 95% confidence limits are based on the usual student t approach using calculated mean and standard deviation

Model(s)		Robust 95% Conf. limits		Student t	mean	s.d	Seductive 95% Conf. limits	
OBS.	mean	369.209	481.772	15.050	425.491	28.271	375.276	481.182
MODEL-A	nmse	0.108	0.245	5.103	0.177	0.035	0.115	0.250
	fb	-0.085	0.091	0.077	0.003	0.044	-0.080	0.091
	corr	0.663	0.887	13.773	0.775	0.056	0.658	0.867
MODEL-B	nmse	0.222	0.469	5.561	0.346	0.062	0.235	0.476
	fb	-0.044	0.158	1.120	0.057	0.051	-0.048	0.150
	corr	0.432	0.764	7.185	0.598	0.083	0.410	0.755
MODEL-C	nmse	0.396	0.761	6.303	0.579	0.092	0.409	0.766
	fb	-0.468	-0.148	-3.841	-0.308	0.080	-0.459	-0.162
	corr	-0.102	0.252	0.848	0.075	0.089	-0.096	0.249

Model(s)		Robust 95% Conf. limits		Student t	mean	s.d	Seductive 95% Conf. limits	
MODEL-A - MODEL-B	nmse	-0.272	-0.066	-3.261	-0.169	0.052	-0.269	-0.074
	fb	-0.125	0.018	-1.489	-0.054	0.036	-0.123	0.015
	corr	0.070	0.284	3.298	0.177	0.054	0.081	0.288
MODEL-A - MODEL-C	nmse	-0.588	-0.216	-4.295	-0.402	0.094	-0.589	-0.219
	fb	0.129	0.494	3.398	0.311	0.092	0.134	0.495
	corr	0.476	0.923	6.228	0.699	0.112	0.454	0.898
MODEL-B - MODEL-C	nmse	-0.416	-0.050	-2.533	-0.233	0.092	-0.415	-0.056
	fb	0.191	0.539	4.167	0.365	0.088	0.210	0.546
	corr	0.274	0.771	4.187	0.522	0.125	0.249	0.745

SUMMARY OF CONFIDENCE LIMITS ANALYSES

D(nmse) among models: an 'X' indicates significantly different from zero

	MODEL-A	MODEL-B	MODEL-C
MODEL-A		X	X
MODEL-B			X

Figure 2. An Example of the Output File for the BOOT Program, Where the Model Performance Measures are based on the C_o and C_p Values Listed in Figure 1. If the User Chooses to Work with the Logarithms of C_o and C_p (Accomplished as a Program Option), then FB will be Replaced with MG, and NMSE will be Replaced with VG.

D(fb) among models: an 'X' indicates significantly different from zero

```
              M   M   M
              O   O   O
              D   D   D
              E   E   E
              L   L   L
              -   -   -
              A   B   C
             -----------
MODEL-A |             X
MODEL-B |             X
```

D(corr) among models: an 'X' indicates significantly different from zero

```
              M   M   M
              O   O   O
              D   D   D
              E   E   E
              L   L   L
              -   -   -
              A   B   C
             -----------
MODEL-A |         X   X
MODEL-B |             X
```

nmse for each model: an 'X' indicates significantly different from zero

```
              M   M   M
              O   O   O
              D   D   D
              E   E   E
              L   L   L
              -   -   -
              A   B   C
             -----------
              X   X   X
```

fb for each model: an 'X' indicates significantly different from zero

```
              M   M   M
              O   O   O
              D   D   D
              E   E   E
              L   L   L
              -   -   -
              A   B   C
             -----------
                      X
```

corr for each model: an 'X' indicates significantly different from zero

```
              M   M   M
              O   O   O
              D   D   D
              E   E   E
              L   L   L
              -   -   -
              A   B   C
             -----------
              X   X
```

Figure 2. An Example of the Output File for the BOOT Program, Where the Model Performance Measures are based on the C_o and C_p Values Listed in Figure 1. If the User Chooses to Work with the Logarithms of C_o and C_p (Accomplished as a Program Option), then FB will be Replaced with MG, and NMSE will be Replaced with VG (Concluded).

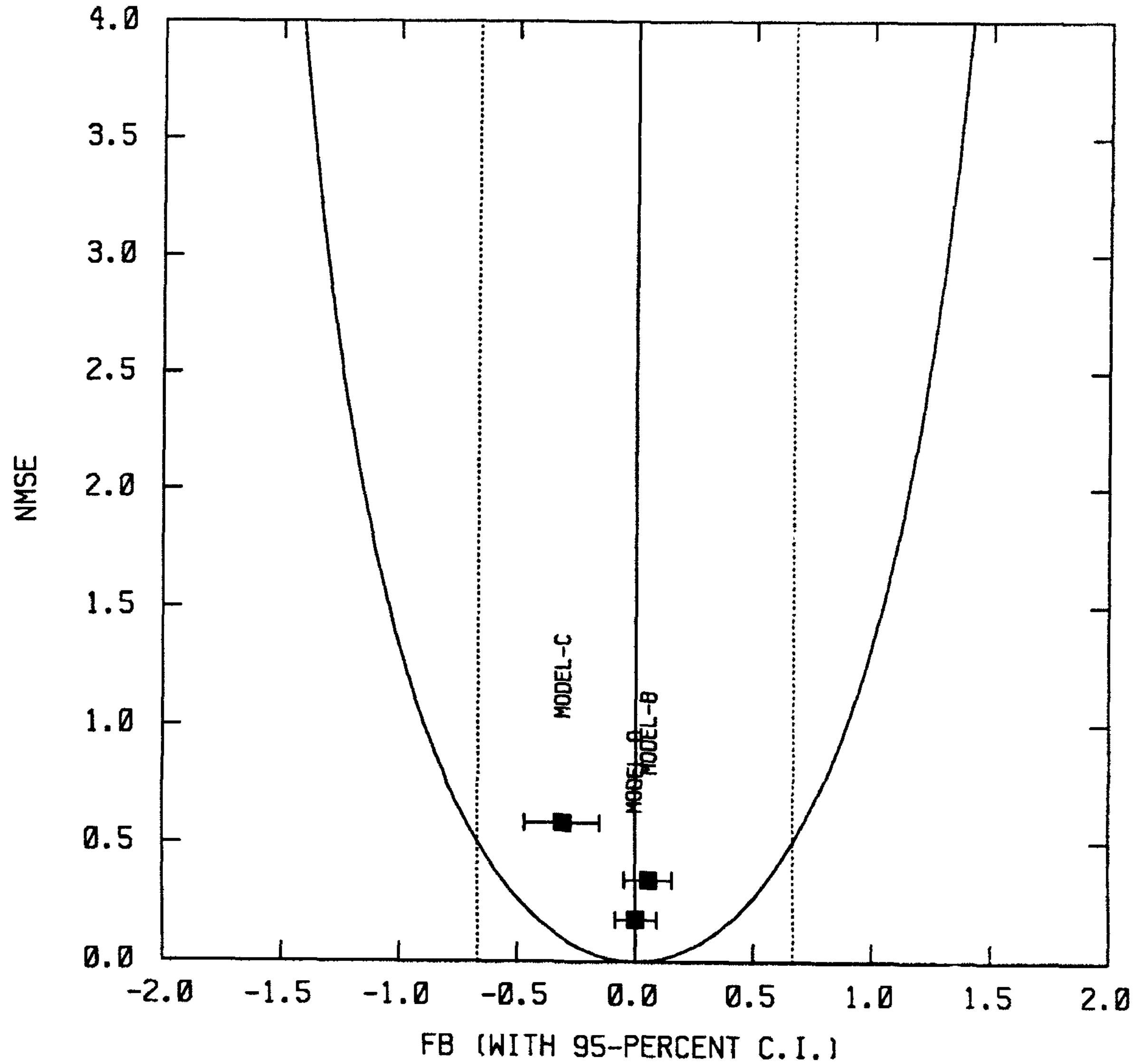


Figure 3. An example of the results generated by the BOOT program and plotted using SIGPLOT, showing the fractional bias FB (with its 95 percent confidence limits) and the normalized mean square error NMSE for each of the models. Also included are the "factor of two" lines (dotted) and the "minimum" NMSE curve (NMSE = $4FB^2/(4 - FB^2)$, solid). Notice that the FB's for Model-A and Model-B are not significantly different from zero, at the 95 percent confidence level.

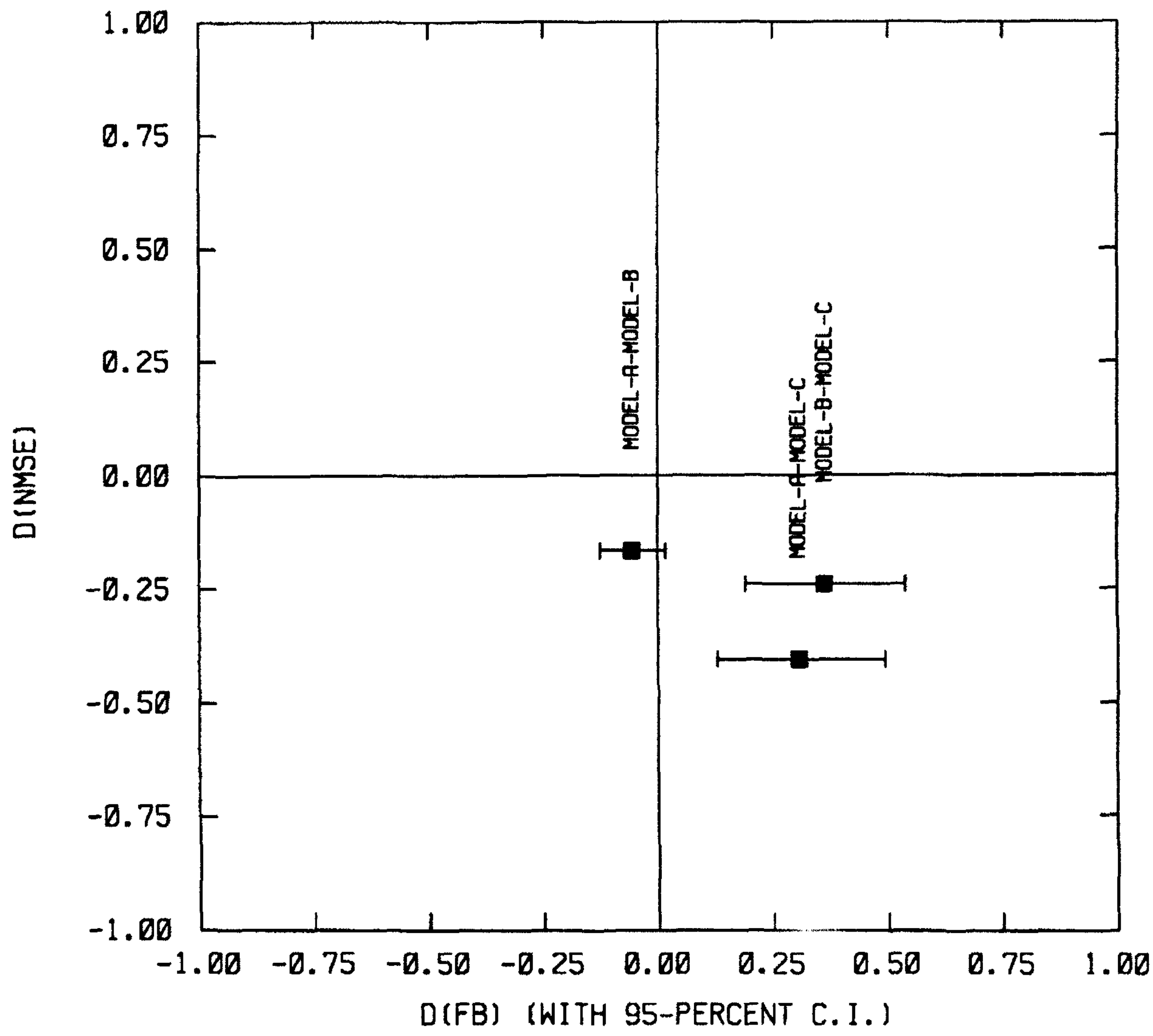


Figure 4. An example of the results generated by the BOOT program and plotted using SIGPLOT, showing the differences in FB (with its 95 percent confidence limits) and NMSE between pairs of models. Notice that the FB for Model-A is not significantly different from the FB for Model-B at the 95 percent confidence level.

- Name of the output file:

The user specifies the name of the mandatory output file here. There is no default answer. The path information for the output file will be implemented, if specified by the user, in a way similar to what was mentioned in the previous question.

- Select one of the following choices from the main program options:

(1) Straight C_o and C_p, with no normalization.
(2) Consider C_o/C_o and C_p/C_o; that is, normalization by C_o
(3) Consider C_o/C_p and C_p/C_p; that is, normalization by C_p
(4) Consider $\ell n(C_o)$ and $\ell n(C_p)$

FB and NMSE will be calculated if "1", "2", or "3" is entered. MG and VG will be calculated if "4" is entered. Option "2" means that both C_o and C_p are normalized by C_o. Option "3" means that both C_o and C_p are normalized by C_p. As described earlier, choice of Option "1" will result in an emphasis on the highest observed and/or predicted concentrations, choice of Option "2" will result in an emphasis on high outliers of C_p/C_o, choice of Option "3" will result in emphasis on high outliers of C_o/C_p, and choice of Option "4" will result in a balanced emphasis over the entire range of observed and predicted concentrations. The user is referred to the end of this section for guidance concerning whether FB and NMSE or MG and VG are preferable. Since the observed data become all ones under Option "2" and the predicted data become all ones under Option "3", the correlation coefficient becomes indeterminate and the fractional variance is always equal to -2.

- Use E- or F-format for mean, sigma, and bias? (e/f):

The user has the option to specify the format used in the mandatory output file. It is suggested that the E-format be used if the magnitude of the input data is large (say, larger than 10000). The default answer (that is, by simply typing the RETURN key) is "f". Note that the G-format is not used here because the decimal points

of the numbers printed out as a result are usually not aligned, thus making the appearance of the output less desirable.

- Do bootstrap resampling? (y/n):

  The BOOT program will carry out the bootstrap resampling procedure and calculate the confidence limits for the performance measures only if the user answers "y" to this question. If this option is chosen, the RANDS file will be opened automatically, and the execution time of the BOOT program will be much longer. The default answer is "y".

The following questions will be asked only if the user answers "y" to the above question.

- Print out detailed information on the confidence limits? (y/n):

  If the user answers "y", the numerical values of the confidence limits for the performance measures will be included in the output file; otherwise, only qualitative results, such as whether FB for a certain model is significantly different from zero, will be included in the output file. The default answer is "y".

- Create files containing FB (including confidence limits) and NMSE, or MG (including confidence limits) and VG which can later be plotted? (y/n):

  Answer "y" if the user wants to plot the results using the SIGPLOT plotting package. The default answer is "y".

The following questions will be asked only if the user answers "y" to the above question.

- Enter name of the file that contains NMSE and FB or VG and MG information:

  The user specifies the name of the file that contains FB (or MG) (with its 95 percent confidence limits) and NMSE (or VG) for each of

the models. There is no default answer. If the user types "TEST.OUT" here, then the results will be written to a file called "TEST.OUT" in the current default directory. If the user types "C:\USR\WIDGET.SUM", then the output will be written to a file called "WIDGET.SUM" in the "C:\USR" directory, assuming that directory already exists.

- Enter name of the file that contains D(NMSE) and D(FB), or D(VG) and D(MG) information:

  The user specifies the name of the file that contains the differences in FB (or MG) (with their 95 percent confidence limits) and NMSE (or VG) between pairs of the models. There is no default answer. The path information for the output file will be implemented, if specified by the user, in a way similar to what was mentioned in the previous question.

It is appropriate to explain the relative merits of the performance measures FB and NMSE versus MG and VG in more detail.

A "perfect" model would have both FB and NMSE equal to 0.0, and both MG and VG equal to 1.0. Geometric mean bias (MG) values of 0.5 and 2.0 can be thought of as "factor of two" overpredictions and underpredictions in the mean, respectively. A geometric variance (VG) value of about 1.6 indicates a typical factor of two scatter between the individual pairs of observed and predicted values.

If there is only a mean bias in the predictions and no random scatter, then the following relations are valid:

$$NMSE = 4FB^2/(4-FB^2) \tag{8}$$

$$VG = \exp[(\ell n MG)^2] \tag{9}$$

As a result, Equation (8) defines the minimum value of NMSE given the value of FB. Equation (9) defines the minimum value of VG given the value of MG .

Note that the performance measures defined in Equation (1) through (7) are merely different measures of the variation of a dataset, which can be used to guide our interpretation of the data. Some measures may be more appropriate than the others depending on the situation at hand.

When there is a large range in values of C_o and C_p in a dataset, the statistics FB, NMSE, and R are very strongly influenced by large values of C_o or C_p. Conversely, small values of C_o and C_p (such as may occur at large distances from a source or in experiments with small source emission rates) do not influence the statistics very much. In this situation, in order to more equally weight the data, the logarithm of concentrations can be taken, and the performance measures MG and VG employed. For example, consider the following data:

C	0.1	0.1	1	10	100
ℓn C	-4.6	-2.3	0	2.3	4.6

As can be seen, when logarithms are taken, underpredictions of a factor of 100 now have the same weight as overpredictions of a factor of 100.

Because the logarithmic forms of the mean bias and variance (MG and VG) are more difficult to visualize than the absolute forms (FB and NMSE), we prefer to use the absolute versions whenever possible. However, use of FB and NMSE is most justified only if there is not a large range in values of C_o and C_p, as described above, and if C_o and C_p are never very different, say, within a factor of two. If a dataset contains several pairs of data with C_o/C_p and C_p/C_o equal to 10, 100, or more, then MG and VG are more appropriate.

To further illustrate, consider the following rather extreme set of data:

		C_o	C_p
Experiment 1	x = 100 m	1100	1000
	x = 1000 m	50	100
Experiment 2	x = 100 m	1.0	10
	x = 1000 m	.01	1

In this example, the source emission rate was much less in Experiment 2 than in Experiment 1. Furthermore, in both experiments, the concentrations decreased by a factor of 10 to 100 between the 100 m and 1000 m arcs. The predictions of the larger concentrations are fairly good, while there are large overpredictions at the smaller concentrations.

The following table contains sets of calculations with these data.

	C_o	C_p	C_o-C_p	$(C_o-C_p)^2$	$\frac{C_o}{C_p}$	$\frac{C_p}{C_o}$	$\ell n\frac{C_o}{C_p}$	$\left(\ell n\frac{C_o}{C_p}\right)^2$	$\frac{C_o}{C_p}-1$	$1-\frac{C_p}{C_o}$	$\left(\frac{C_o}{C_p}-1\right)^2$	$\left(1-\frac{C_o}{C_p}\right)^2$
	1100	1000	100	10000	1.1	.91	.095	.01	.10	.09	.01	.01
	50	100	-50	2500	0.5	2.0	-.693	.48	-.50	-1.0	.25	1
	1.0	10	-9	81	0.1	10.0	-2.3	5.29	-.90	-9.0	.81	81
	.01	1	-.99	.98	0.01	100.0	-4.61	21.25	-.99	-99.0	0.98	9801
avg.	287.75	277.75	10	3145.5	0.43	28.23	-1.88	6.75	-.57	-27.3	.51	2470

These data lead to the following performance measures

Measures of Mean Bias

$$FB_1 = (\overline{C_o - C_p})/0.5(\bar{C}_o + \bar{C}_p) = 0.035$$

$$FB_2 = (1 - \overline{C_p/C_o})/0.5(\bar{C}_p/\bar{C}_o + 1) = -1.86$$

$$FB_3 = (\overline{C_o/C_p} - 1)/0.5(\bar{C}_o/\bar{C}_p + 1) = -0.80$$

$$\ell n\ MG = (\overline{\ell n(C_o/C_p)}) = -1.88$$

Measures of Variance

$$NMSE_1 = \overline{(C_o - C_p)^2}/(\bar{C}_o\bar{C}_p) = 0.039$$

$$NMSE_2 = \overline{(1 - \overline{C_p/C_o})^2}/(\overline{C_p/C_o}) = 87.5$$

$$NMSE_3 = \overline{(C_o/C_p - 1)^2}/(\overline{C_o/C_p}) = 1.19$$

$$\ell n\ VG = \overline{(\ell n(C_o/C_p)^2} = 6.75$$

Subscript 1 represents the FB and NMSE measures defined by Equations (1) and (3). Subscripts 2 and 3 represent FB and NMSE calculated after concentrations are normalized by C_o and C_p, respectively.

The four alternate performance measures have the following emphasis:

FB_1 and $NMSE_1$: Emphasis on high observed and/or predicted concentrations

FB_2 and $NMSE_2$: Emphasis on high outliers of C_p/C_o (that is, large overpredictions, independent of magnitude)

FB_3 and $NMSE_3$: Emphasis on high outliers of C_o/C_p (that is, large underpredictions, independent of magnitude)

ℓn MG and ℓn VG: Balanced Emphasis

Note that FB_1 and $NMSE_1$ will be calculated if the main program Option "1" (described earlier) has been chosen, FB_2 and $NMSE_2$ will be calculated if the main program Option "2" has been chosen, FB_3 and $NMSE_3$ will be calculated if the main program Option "3" has been chosen, and MG and VG will be calculated if the main program Option "4" has been chosen.

SECTION VIII

SCIENTIFIC EVALUATION OF MODELS USING RESIDUAL PLOTS

One way of evaluating the scientific credibility of a model is through the use of residual plots, where "residual" is defined as the ratio of the predicted to the observed concentration (note that the logarithm of this ratio equals the difference between the logarithm of the two concentrations). Values of the residual can be plotted versus values of variables such as wind speed or stability. The residuals of a good model (1) should not exhibit any trend with variables such as wind speed and stability class, and (2) should not exhibit large deviations from unity (implying a perfect match between the model and the observed). The SIGPLOT plotting package (see Appendix A) is used to generate the residual plots. The RESIDUAL program, described below, is used to generate the special input file required by SIGPLOT from a file containing multiple columns of data, representing concurrent values of the observations, model predictions, and other primary variables such as wind speed and stability. The RESIDUAL program is written in FORTRAN 77.

In the RESIDUAL program, the user first defines certain ranges of the primary variables in the input file to be used for grouping the residuals and plotting them by means of "box plots." Grouping is usually necessary because of the large number of data points. The cumulative distribution function (cdf) of the residuals within each group is represented by the 2nd, 16th, 50th, 84th, and 98th percentiles. These five significant points in the cdf are then plotted by the SIGPLOT program using a "box" pattern. As mentioned above, it is desirable that the residual boxes of a model should not exhibit any systematic dependence on the primary variables. It is also desirable that the residual boxes should be compact and should not deviate too much from unity.

The RESIDUAL program requires one input file and generates one output file. The output file then serves as the input file to the SIGPLOT plotting package. There are no default names associated with these files, and the user is prompted for the file names during the execution of the program.

The input data file of the RESIDUAL program contains multiple columns of data, representing concurrent values of the observations, model predictions, and other primary variables such as wind speed and stability. The ranges of

the primary variables are also defined, to be used for grouping the data. Table 2 describes the format of the input file. Figure 5 shows an example of the input file. The input data file accepted by the RESIDUAL program can also be accepted by the BOOT program, but not vice versa. It is recommended that the user always prepare the input data file according to the format described in Table 2 so that both the RESIDUAL and BOOT programs can be executed using the same input file. Note that the RESIDUAL program makes no corrections or substitutions for missing data; it is the responsibility of the user to provide valid data at each position.

The output file of the RESIDUAL program contains distributions (the 2nd, 16th, 50th, 84th, and 98th percentiles of the cdf) of the residuals as a function of the primary variables. The information stored in this output file can then be plotted using the SIGPLOT plotting package (see Figure 6 for an example).

During the execution of the RESIDUAL program, the following questions will be asked:

- Name of the input file:

  The user must specify the name of the input data file here. There is no default answer. For example, if the input data are contained in a file called "TEST.DAT" residing in the current default directory, then the user should type "TEST.DAT" here. If the input data are contained in a file called "WIDGET.INP" residing in a directory called "C:\USR", then the user should type "C:\USR\WIDGET.INP" here.

- Name of the output file:

  The user must specify the name of the output file. There is no default answer. The path information for the output file will be implemented, if specified by the user, in a way similar to what was mentioned in the previous question.

TABLE 2. FORMAT OF THE MANDATORY INPUT DATA FILE OF THE RESIDUAL PROGRAM. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL.

LINE NO.	FORMAT	DESCRIPTION
1	FF/I	There are four integer constants in this line, representing the total number of observations (NN, < 501), the total number of models (MM, < 16, including the observation (or some baseline model) as one, the total number of blocks (KK), and the total number of primary variables (NVAR, < 11). Note that KK is in fact not used by RESIDUAL since the blocking of data is performed internally according to the defined ranges of the primary variables. However, KK is used by the BOOT program. The limits on NN, MM, and NVAR are assigned in the program using the PARAMETER statements, and can be easily changed.
2	FF/I	The information in this line is not used by the RESIDUAL program. If the same input file is to be read by the BOOT program, there should be KK integer constants in this line, representing the number of pieces in each block. Note that the sum of all these integers must equal NN.
3	FF/I	There are MM character constants in this line; each one can be at most eight characters long, containing the name of each of the models. All character constants must be enclosed in apostrophes.
4	FF/I	The information in this line is not used by the RESIDUAL program. If the same input file is to be used by the BOOT program, there should be KK character constants in this line. Each one can be at most 20 characters long, containing the name of each of the blocks. All character constants must be enclosed in apostrophes.

TABLE 2. FORMAT OF THE MANDATORY INPUT DATA FILE OF THE RESIDUAL PROGRAM. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL (CONCLUDED).

LINE NO.	FORMAT	DESCRIPTION
Next NN lines:		
	FF/R	There are MM+NVAR real numbers in each line, with the first number representing the observed value (or the prediction based on some base line model), the following MM-1 numbers representing the prediction from each of the remaining models, and the following NVAR numbers representing each of the primary variables.
Next NVAR lines:		
	FF/ I,C,R	Each line describes the way each of the NVAR primary variables is to be blocked. The first parameter is an integer (IXR, < 21), representing the number of ranges for the primary variable. The second parameter is a character constant, at most 40 characters long, enclosed in apostrophes, representing the name of the primary variable. The next IXR+1 real numbers, in numerical ascending order, define the boundaries of the ranges. For example, the following line: 4 'u (m/s)' 0. 2. 5. 10. 20. means that wind speeds should be divided into four groups where the distribution of the model residuals within each group is to be calculated. The first group is for those data when wind speeds are between 0. and 2. m/s, the second group is for wind speeds between 2. and 5. m/s, etc. The limits on IXR are assigned in the program using the PARAMETER statement, and can be easily changed. Note that the sequence of the NVAR lines must be consistent with that of the last NVAR columns described in the previous section. As an example, if the MM+1th column in the previous section contains information for wind speeds, then the first line in this section should also contain grouping information for wind speeds.

```
  79     4  2  4
  39 40
'OBS.' 'MODEL-A' 'MODEL-B' 'MODEL-C'
'Urban data set' 'Rural data set'
  616.0   708.7   594.7   516.5     11     3.0    800.  2
  604.1   689.2   585.8   496.7     12     3.4   1000.  2
  868.0   674.8   580.3   516.8     13     3.5   1100.  2
  498.6   668.8   652.1   548.3     14     3.8   1200.  2
  393.1   560.2   704.7   581.9     15     4.7   1300.  2
  409.0   740.9   570.1   621.4     16     5.2   1000.  3
  640.2   249.6   510.1   553.5     17     5.4   1100.  3
  265.3   259.6   463.4   446.0     18     4.9   1100.  4
  192.7    91.6   131.0   485.0     19     4.2   1100.  5
 1149.1  1217.5  1116.1   520.6     10     2.6   1600.  2
  972.8  1275.8  1175.1   536.9     11     3.2   1900.  2
 1137.5  1225.7  1081.7   617.4     12     3.8   1600.  2
  669.5  1052.8   905.1   637.3     13     4.5   1600.  2
  595.5   862.0   862.0   664.1     14     5.0   1500.  2
  741.2   589.5   767.0   665.3     15     5.1   1500.  2
  612.6   602.4   728.2   672.4     16     5.0   1500.  3
  312.0   398.9   657.5   659.5     17     5.2   1500.  3
  400.2   340.2   412.3   586.0     18     5.1   1500.  4
  264.7   612.1   774.2   705.9     16     5.7   1400.  3
  290.0   428.4   757.3   708.8     17     5.1   1800.  3
  459.5   355.0   512.3   602.4     18     5.1   2000.  4
  444.0   216.0   441.4   681.1     19     4.4   2000.  5
  175.1   216.6   456.1   825.4     20     4.6   2000.  6
  102.3   126.1   255.6   522.9     21     4.9   2000.  6
  128.8    16.5     0.5   834.9     22     4.6      0.  6
  200.2   301.9   208.9   728.0     23     5.4      0.  6
  358.3   481.8   354.0   742.4     24     5.4      0.  6
  611.1  1010.2   987.1   679.0     14     4.4   1500.  2
  499.3   752.5   921.6   725.7     15     5.0   1500.  2
  537.8   724.0   826.8   675.9     16     4.7   1500.  3
  220.0   523.3   908.2   640.8     17     3.9   1800.  3
  479.2   357.5   788.6   544.7     18     4.2   2000.  4
  133.2   195.3   383.1   738.5     19     3.1   1800.  5
   98.2   167.3   213.5  1064.9     20     3.2   1500.  6
   92.5   104.6   142.2   741.2     21     3.1   1200.  6
   21.0   127.4   176.3   805.2     22     3.3   1200.  6
  353.0   307.8   167.1   576.9     20     3.8   2000.  5
  358.0   280.9   188.4   225.3     21     2.3   2000.  4
  233.3   355.3   234.9   719.1     22     2.4   2000.  5
  198.3    12.7   184.0   745.2     23     3.6   2000.  6
```

Figure 5. An example of the Input Data File for the RESIDUAL Program. Refer to Table 2 for the Format of the File. In Brief, the Example Shows that there are 79 Data Points from 4 Models, under the Names "OBS.", "MODEL-A", "MODEL-B", and "MODEL-C", respectively. The 79 Data Points are Divided into 2 Blocks: the first Block, "Urban Data Set", includes the first 39 Points; the second Block, "rural data sets", includes the next 40 Points. Note, however, that this Blocking Information is not used by the RESIDUAL Program. The Reason that this Information is Retained is that the same File Listed here can also Serve as Input to the BOOT program (see Figure 2). The Blocking is Achieved according to the 4 Primary Variables: Hour of Day, Wind Speed, Mixing Height, and Stability Class. The Main Section of the Input File Lists the Concurrent Values of the 4 Models and the 4 Primary Variables. The Last Portion of the File Describes the Blocking Information for each of the Primary Variables. For Example, 10 Blocks in Wind Speeds, u, are Considered, $0.5 \le u < 1.5$ is Considered as One Block, $1.5 \le u < 2.5$ is Considered As Another Block, etc.

```
  507.2    0.0  126.3  664.9    24    3.5  2000.  6
  313.7    0.0    0.0  667.1     1    4.2     0.  6
  165.1    0.0    0.0  703.9     2    3.6     0.  6
  295.6  329.9  454.6  695.3     4    5.2     0.  6
  527.7  308.0  295.9  775.0     5    4.7     0.  6
  454.1  301.0    1.0  995.6     6    2.9     0.  6
  240.3  417.5  361.1  933.8     7    3.4     0.  6
  590.8  579.3  144.2  666.5     8    3.1  1500.  5
  638.3  756.6  608.9  400.1     9    3.4  1500.  4
  949.8 1004.2  805.4  528.9    10    3.4  1500.  3
  886.8  855.6  706.2  517.4    11    3.0  1300.  2
  635.5  761.0  670.9  596.6    12    4.5  1200.  2
  359.3  412.6  232.5  937.6     1    2.3  1200.  6
  484.7  360.7  226.8  979.0     2    2.5  1200.  6
  529.7  332.0  202.5  980.0     3    2.4  1200.  6
  585.8  291.4  186.1 1100.1     4    2.1  1200.  6
  367.7  368.0  260.2 1005.6     5    2.1  1200.  6
  324.7  270.9   72.7 1058.6     6    2.0  1200.  6
  489.0  274.6  208.5  942.2     7    2.6  1200.  6
  570.8  337.1  218.0  646.5     8    2.8  1200.  5
  419.7  254.4  206.1  344.0     9    4.3  1200.  4
  532.8  414.2  197.9  477.0     9    4.8  1800.  3
  425.2  365.7  198.7  469.5    10    7.1  1700.  4
  467.5  411.5  228.5  455.3    11    7.5  2000.  4
  362.2  306.4  147.6  405.2    12    5.1  2000.  4
  429.2  287.4  139.2  450.6    13    5.4  2000.  4
  446.0  338.1  169.5  461.2    14    5.7  2000.  4
  192.9  253.8  145.6  460.7    15    5.8  2400.  4
  630.3  322.5  257.2  460.5    16    7.3  2700.  4
  364.9  326.7  251.1  510.6    17    7.8  3000.  4
  111.4  196.4  248.5    0.0    23    1.9   250.  4
   89.8  146.5  254.9    0.0    24    1.9   250.  4
   82.5  248.0  160.9    0.0     1    2.9   250.  4
  296.5  253.2  193.2    0.0     2    2.8   250.  4
  215.4  299.7  165.0    0.0     3    2.9   250.  4
  454.5  274.2  154.0    0.0     4    3.5   250.  4
  384.7  324.6  163.2    0.0     5    3.5   250.  4
  253.2  488.3  175.6    0.0     6    3.5   250.  4
  289.5  304.1  193.1    0.0     7    3.1   250.  4
6 'hour of day' -0.01, 4., 8., 12., 16., 20., 24.01
10 'u (m/s)' 0.5, 1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5, 10.5
6 'h (m)' -0.01, 200., 600., 1000., 1500., 2000., 3000.1
3 'pg class' 0.5 3.5 4.5 6.5
```

Figure 5. An example of the Input Data File for the RESIDUAL Program. Refer to Table 2 for the Format of the File. In Brief, the Example Shows that there are 79 Data Points from 4 Models, under the Names "OBS.", "MODEL-A", "MODEL-B", and "MODEL-C", respectively. The 79 Data Points are Divided into 2 Blocks: the first Block, "Urban Data Set", includes the first 39 Points; the second Block, "rural data sets", includes the next 40 Points. Note, however, that this Blocking Information is not used by the RESIDUAL Program. The Reason that this Information is Retained is that the same File Listed here can also Serve as Input to the BOOT program (see Figure 2). The Blocking is Achieved according to the 4 Primary Variables: Hour of Day, Wind Speed, Mixing Height, and Stability Class. The Main Section of the Input File Lists the Concurrent Values of the 4 Models and the 4 Primary Variables. The Last Portion of the File Describes the Blocking Information for each of the Primary Variables. For Example, 10 Blocks in Wind Speeds, u, are Considered, $0.5 \le u < 1.5$ is Considered as One Block, $1.5 \le u < 2.5$ is Considered As Another Block, etc. (Concluded).

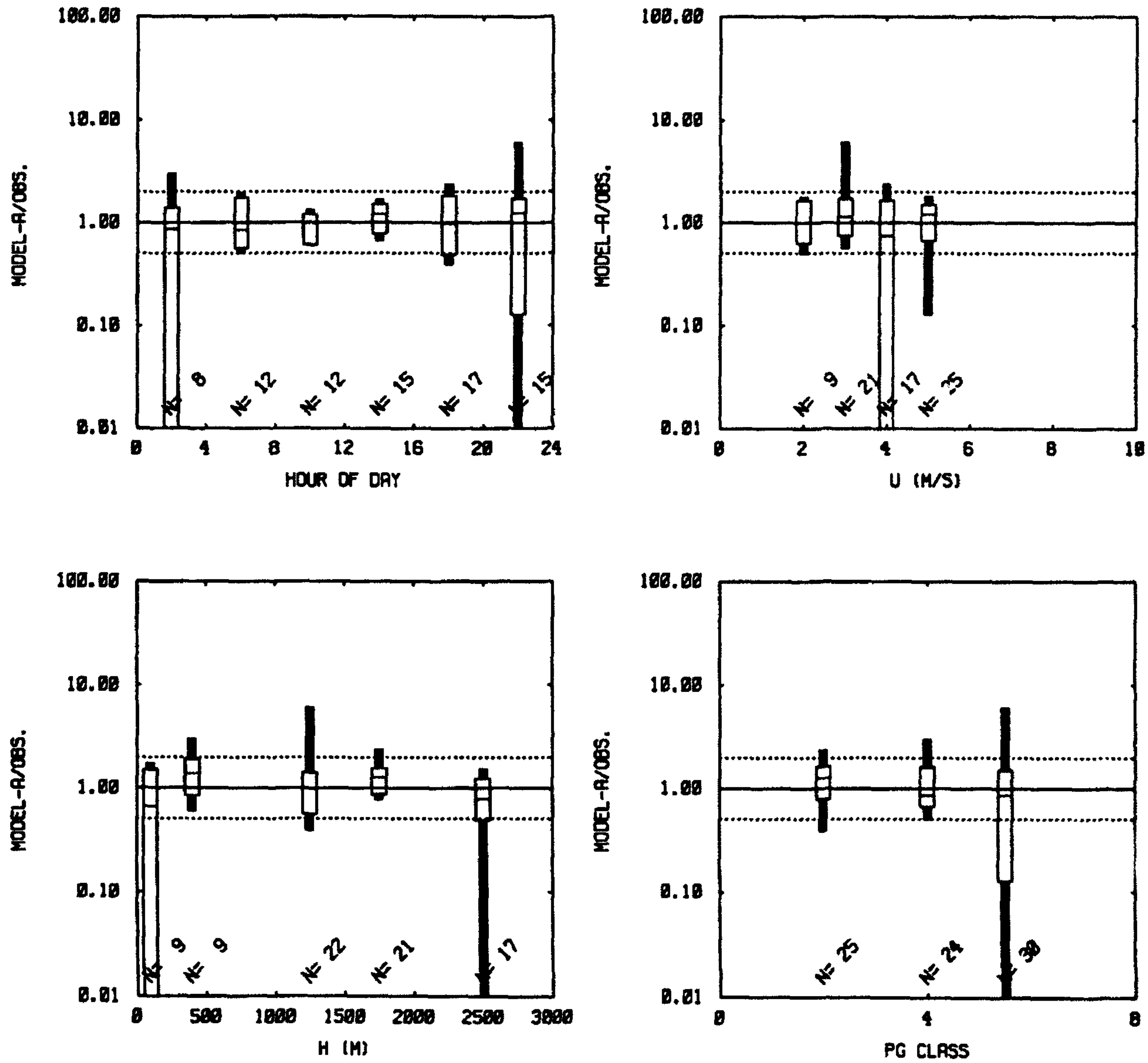


Figure 6. An Example of the Results Generated by the RESIDUAL Program and Plotted using SIGPLOT. Significant Points on each Box Plot Represent the 2nd, 16th, 50th, 84th, and 90th Percentiles. The Numbers of Observations used in each Box are also Labeled near the Bottom as "N = #." The Dashed Lines Represent the Factor-of-Two Lines.

- An input file typically contains several columns of data. In most cases, these columns represent concurrent values of the observations, model predictions, and other primary variables such as wind speed and stability. In the sample case listed in Figure 5, for example, the main section of the input file has eight columns. Column 1 represents the observed concentration values, and columns 2 through 4 represent the predicted concentrations by "MODEL-A", "MODEL-B", and "MODEL-C", respectively. The remaining four columns represent the following four primary variables: hour of day, wind speed, mixing height, and stability class. The RESIDUAL program treats only the ratio of any two dependent variables (that is, columns 1 through 4 for the example in Figure 5), specified by the user, at a time.

- Implement a lower threshold on the ratio? (y/n):

  The user has the option of specifying a lower threshold for the ratio of the two columns of numbers chosen above. This is sometimes necessary if the logarithmic scale is to be used and one of the columns has zero or minute values. The default (that is, hitting the RETURN key) answer is "y".

The following request will be made only if the user answers "y" to the above question.

- Enter the lower threshold of the ratio (for example, 0.01):

  The user specifies the value of the lower threshold of the ratio. There is no default answer, but 0.01 has proven to be a good choice for C_p/C_o in our tests.

SECTION IX

ESTIMATION OF MODEL UNCERTAINTY COMPONENTS

A. UNCERTAINTY DUE TO DATA ERRORS--MONTE CARLO SENSITIVITY ANALYSES

Model sensitivity to input data errors can be investigated using the Monte Carlo method. The method involves running a model multiple times, with the input parameters slightly perturbed each time. It is necessary to implement the Monte Carlo sensitivity analyses on a computer platform where the user can easily run the model repeatedly, can efficiently extract the information of interest, and can not be overwhelmed by the amount of output generated. The following procedures for implementing the Monte Carlo sensitivity analyses are recommended:

Step 1: Choice of Model

Since it is necessary to run a model hundreds to thousands of times, there are some important criteria for choosing a dispersion model for application of the Monte Carlo sensitivity analyses. First, it is desirable that the input, the execution and the postprocessing of the model be fully automated. Second, it is desirable that the model can execute reasonably quickly. Last, a somewhat less stringent requirement, the model should have a simple I/O structure, such as small numbers of compact input and output files.

Step 2: Choice of Input Parameters to be Perturbed

Input parameters accepted by the models can be classified as primary and secondary. Only the primary input parameters should be involved in the Monte Carlo procedure. Secondary input parameters are derived from the primary input parameters. The new values of the secondary input parameters are assumed to be derived from the updated primary input parameters based on known physical relationships. Examples of primary input parameters are: wind velocity, temperature, surface roughness, source emission rate, and source dimension. Examples of secondary input parameters are Monin-Obukhov length, friction velocity, and stability class. The user should exercise careful judgment in selecting the input parameters to be analyzed. For example, for those models (such as DEGADIS) that treat the secondary source blanket, the

source emission rate is expected to be an important input parameter. However, for those models (such as GPM) where the predicted concentration is proportional to the source emission rate, model sensitivity to the error in the source emission rate can be easily predicted without resorting to the Monte Carlo analyses.

After choosing the primary input parameters, the user needs to specify a form for the probability distribution function (pdf) from which random samples will be drawn. Optional pdf's are: uniform, exponential, Gaussian, log-normal, and clipped normal. In the clipped normal pdf, any negative tail in the Gaussian pdf is replaced with a delta function at zero (Reference 5).

It is also necessary to estimate the mean and variance (or uncertainty) of the primary input parameters. The variances or uncertainties associated with meteorological observations depend on the kind of the instrument used, the averaging time, the orientation with the wind direction, and the atmospheric stability (see Volume III of this report). The uncertainties associated with other parameters, on the other hand, can be estimated by some simple rules of thumb. For example, the uncertainty in the estimate of surface roughness can be considered as large as one order of magnitude. That is, if the reported value of surface roughness is 0.3 cm, then any values between 0.1 and 1.0 cm are possible.

Step 3: Method for Randomly Choosing a Number from a Given pdf

The following functions or subroutines are available to help the user to randomly select a number from the uniform, exponential, Gaussian, log-normal, and clipped normal distributions. All distributions rely on a uniform random number generator program, which is readily available either as a system routine or through various textbooks on numerical methods (Reference 6.)

- RAN1 (IDUM)

The RAN1 subroutine returns a uniform random number, R_0, between 0.0 and 1.0. By setting IDUM equal to any negative number, the random number sequence is initialized. A uniform random number R_1 between y_u and y_l ($y_u > y_l$) is created using the following equation:

$$R_1 = R_0(y_u - y_l) + y_l \tag{10}$$

where the mean (μ) and standard deviation (σ) of R_1 are:

$$\mu = (y_u + y_l)/2 \tag{11}$$

$$\sigma = ((y_u^2 + y_u y_l + y_l^2)/3 - \mu^2)^{1/2} \tag{12}$$

The RAN1 routine was adopted directly from Press et al. (Reference 6).

- EXPDEV (IDUM,BETA)

The EXPDEV subroutine returns an exponentially distributed, positive, random number whose mean and standard deviation are both equal to BETA (> 0). RAN1 is used as the source of uniform random numbers. By setting IDUM equal to any negative number, the random number sequence is initialized. The EXPDEV routine was adopted from Press et al. (Reference 6), with a slight modification.

- GASDEV (IDUM)

The GASDEV subroutine returns a normally distributed random number, R_0, with zero mean and unit standard deviation. RAN1 is used as the source of uniform random numbers. By setting IDUM equal to any negative number, the random number sequence is initialized. A normally distributed random number, R_1, with a mean of μ and a standard deviation of σ can be obtained using the following equation:

$$R_1 = \mu + \sigma R_0 \tag{13}$$

The GASDEV routine was adopted directly from Press et al. (Reference 6).

• RLNODEV (IDUM,XM,S)

The RLNODEV subroutine returns a log-normally distributed random number, using RAN1 as the source of uniform random numbers. The parameter XM controls the mean of the log-normal pdf, and the parameter S is roughly equal to the ratio of the standard deviation to the mean of the pdf. Both XM and S can be either positive or negative; however, the values of the random samples will always be positive. The magnitude of S should be less than one in order to assure the accuracy of the program. By setting IDUM equal to any negative number, the random number sequence is initialized. The mean and standard deviation of the log-normal distribution are:

$$\mu = \exp(XM + S^2/2) \tag{14}$$

$$\sigma = \exp(2XM + 2S^2) - \exp(2XM + S^2) \tag{15}$$

The RLNODEV routine was derived from the GASDEV routine.

• CL2NPAR (XA,RAT,GBAR,SIGG)

The CL2NPAR subroutine returns the mean and standard deviation (GBAR and SIGG) for an equivalent Gaussian distribution, given the ratio (RAT) of the standard deviation to the mean for a clipped normal distribution and the lower threshold of sampling (XA). Note that GBAR, SIGG and XA all represent values normalized by the mean of the clipped normal distribution. The value of XA must be between -1 and 1. The value of RAT should be between 0.1 and 2 to have the best results.

To illustrate the usage of CL2NPAR, suppose a user wants to construct a clipped normal distribution with mean, standard deviation and lower threshold equal to 2., 2., and 0.4, respectively. In this case, RAT = 2./2. = 1., and XA = 0.4/2. = 0.2. The CL2NPAR routine then returns a value of 0.5157 for GBAR and 1.5405 for SIGG. In other words, if the user selects many samples from a Gaussian distribution with the mean and standard deviation equal to 0.5157*2 = 1.0314, and 1.5405*2 = 3.0810, respectively, and resets the value of the sample to 0.4 whenever it is smaller than 0.4, then the mean and

standard deviation of these samples will be roughly equal to 2. and 2., respectively, as originally specified by the user. CL2NPAR itself does not perform any sampling, the actual sampling is performed by the GASDEV routine after GBAR and SIGG are returned from CL2NPAR.

The CL2NPAR routine is a generalization of the CLNPAR routine included in the SCIMP (Second-order Closure Integrated Model Plume) model code developed by the ARAP Division of California Research and Technology (Reference 7). CL2NPAR reduces to CLNPAR if XA equals zero.

Among the five distributions described above, the uniform distribution is bounded at both ends, the exponential, log-normal and clipped normal distributions are bounded at one end, and the Gaussian distribution is unbounded. Except for the uniform distribution, some precautions should be taken to prevent physically unreasonable outliers (for example, a wind speed of 75 m/s) from being sampled.

Finally, a FORTRAN program, TESTSAMP, is written to test the validity of the above five sampling routines, and to help the user to familiarize himself with the way that these routines are actually implemented.

Step 4: Code Package for Running a Model Repeatedly

Two possible approaches can be taken to carry out repeated runs of a model. First, the user can write a master driver program that implements (1) random sampling of the input parameters, and (2) running of the model. The user needs to place the model in a subroutine where new samples of the input parameters (generated elsewhere within the master driver program) are passed into the model during each Monte Carlo simulation. The advantage of this approach is that, since the Monte Carlo simulations of the model are implemented within a master driver program, the results for each simulation can be easily integrated and later analyzed. At the same time, the normal output of the model should be disabled so that only the required information is saved (for example, a few concentration values). This approach relies on the availability of the source code of the model. Unfortunately, modification of the source codes is not a trivial task, since the I/O structure of the model must be changed, requiring a working knowledge of the model.

As a second approach, the user can write a program that generates hundreds or thousands of versions of the input file of the model, prior to running the model. The Monte Carlo simulations of the model can then be accomplished using a DOS batch file. It is suggested that the user also code a post-processor program to extract only the required information from the outputs, since it is not practical to process the results manually each time. This approach does not involve the modification of the model. However, additional efforts are needed to incorporate the results of each simulation into a file to be later processed.

The user is referred to Volume II of this report for an example of the application of the Monte Carlo sensitivity analysis procedure to five dispersion models (AFTOX, DEGADIS, GASTAR, GPM, and SLAB). This particular implementation combines the advantages of the above two approaches in that all the procedures, including the sampling of the input parameters, the execution of the models, and the post-processing of the results, are integrated into a single software package. No modifications of the models were required in this particular exercise. This is made possible mainly through the use of the DOS interface subroutine, an extension of the Lahey FORTRAN compiler.

Step 5: Analysis of Results

The following is a list of analyses that should be performed after obtaining the results from all Monte Carlo simulations.

- Analyze the range of the model predictions as a function of the ranges of input parameters.

- Analyze the relative model variation, $(\sigma_C/\bar{C})^2$, as a function of relative input data error, $(\sigma_I/\bar{I})^2$, where C is the model output (such as concentration) and I is an input parameter (such as wind speed). When more than one input parameter is perturbed, the question of whether whether the model error is roughly equal to the sum of the input data errors associated with each parameter should be investigated.

- Analyze the pdf of the model results in order to determine if it agrees with the assumed pdf for the input parameters.

- Determine whether one can obtain just as much information from a very simple type of sensitivity analysis in which two model runs are made--one for $\bar{I} - \sigma_I$ and one for $\bar{I} + \sigma_I$. Typically, this would be the case if the model varies with the input data in an approximately linear fashion, such as the relation of the source emission rate to the concentration predicted by the AFTOX model.

B. STOCHASTIC COMPONENT OF UNCERTAINTY

According to Hanna (Reference 8), the stochastic component of uncertainty, σ/μ, depends on downwind distance (x), crosswind distance from the centerline (y), vertical distance from the centerline (z), integral time scale of concentration fluctuations (T_c), dispersion coefficients (σ_{yT} and σ_{zT}), Lagrangian turbulence length scales (T_{Ly} and T_{Lz}), concentration averaging time (T_a), concentration averaging distance (D_a), and initial size of the source (σ_0). Volume III of this report describes the theoretical background in more detail. The theoretical equations relating σ/μ to various variables were developed from experiments involving continuous releases of non-buoyant gases from point sources, and are not fully validated for other possible meteorological and source conditions. Even under ideal conditions, estimation of values of the input variables (for example, T_{Ly} and T_{Lz}) is not an easy task.

The ESTSIG subroutine returns the value of σ/μ, given the input variables x, y, z, T_c, σ_{yT}, σ_{zT}, T_{Ly}, T_{Lz}, T_a, D_a, and 0. In order to assure the robustness of the algorithm for estimating σ/μ, only the most practical formulas were chosen to implement in ESTSIG. It is assumed that σ/μ depends on y, z, σ_{yT}, σ_{zT}, T_a, and T_c through the following two equations:

$$\sigma_c/\bar{C} = \exp\left(\frac{y^2}{4\sigma_{yT}^2}\right)\exp\left(\frac{z^2}{4\sigma_{zT}^2}\right) \tag{16}$$

$$\frac{\sigma_c^2(T_a)}{\sigma_c^2(0)} = \left(1 + \frac{T_a}{2T_c}\right)^{-1} \tag{17}$$

Therefore, in this preliminary version of the code, even though ESTSIG

requires the input of x, T_{Ly}, T_{Lz}, D_a, and 0, their values do not influence the result. New formulas relating σ/μ to these parameters will be added to the code when they are further validated.

The following procedures have been taken to further ensure the robustness of the above two formulas: (1) when either of the dispersion coefficients, σ_{yT}, and σ_{zT}, is missing, it is assumed that it has negligible correction to σ/μ, (2) the magnitude of y/σ_{yT} and z/σ_{zT} is not allowed to exceed 2.0, and (3) T_c is always assumed to be equal to 100 seconds regardless of the user input.

C. TOTAL UNCERTAINTY

The procedures recommended in the two subsections above are capable of producing estimates of two components of the total model uncertainty--the components due to data input errors and due to stochastic fluctuations. A third, unknown, component is the contribution due to model physics errors, which cannot be quantitatively estimated at this time, and is not included in the estimate of the total model uncertainty.

Ignoring the uncertainties due to model physics errors, estimate of the minimum total model uncertainty is given by the formula:

$$\frac{\overline{\left(C_o - C_p\right)^2}}{\bar{C}_o\bar{C}_p} = \frac{\sigma_c^2}{\bar{C}^2}\text{ (data errors)} + \frac{\sigma_c^2}{\bar{C}^2}\text{ (stochastic)} \tag{18}$$

where $\sigma_c{}^2/\bar{C}^2$ (data errors) is obtained from the sensitivity analysis (Section IX A) and $\sigma_c{}^2/\bar{C}^2$ (stochastic) is obtained from Section IX B.

It should be mentioned that, whenever we have estimated the total uncertainty using Equation (14), its magnitude has been found to be much larger than the true value of $\overline{(C_o - C_p)^2}/\bar{C}_c\bar{C}_p$ calculated from the experimental data using the software in Section VII. There appear to be compensating errors that are not well-known.

REFERENCES

1. Computer Sciences Corp., Bootstrap User's Manual (A Method for Comparing Model Performance), Version 1.0, Prepared by U.S. EPA, Office of Air Quality Planning and Standards, Research Triangle Park, NC, 27711, Prepared by Computer Sciences Corp., 79 T.W. Alexander Dr., Research Triangle Park, NC 299090, 1991.

2. Cox, W.M. and Tikvart, J.A., "Statistical Procedure for Determining the Best Performing Air Quality Simulation Model," Atmos. Environ., pp. 2387-2395, 1990.

3. Efron B., "Better Bootstrap Confidence Intervals," J. Amer. Statist. Assoc., vol 82, pp. 171-185, 1987.

4. Hanna, S.R., "Confidence Limits for Air Quality Models, as Estimated by Bootstrap and Jackknife Resampling Methods," Atmos. Environ., pp. 1385-1395, 1989.

5. Lewellen, W.S. and Sykes, R.I., "Meteorological Data Needs for Modeling Air Quality Uncertainties," J. Atmos. and Oceanic Tech., pp. 759-768, 1989.

6. Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T., Numerical Recipes, The Art of Scientific Computing, Cambridge University Press, 818 pp., 1986.

7. Parker, S.F., Sykes, R.I., Henn, D.S. and Lewellen, W.S., User's Manual for SCIMP, Submitted by ARAP Division of California Research & Technology, Inc., 50 Washington Road, P.O. Box 2229, Princeton, NJ to Electric Power Research Institute, 3412 Hillview Ave., Palo Alto, CA, 1989.

8. Hanna, S.R., "Concentration Fluctuations in a Smoke Plume," Atmos. Environ., pp. 1091-1106, 1984.

APPENDIX A

A. USER'S GUIDE FOR THE SIGPLOT PLOTTING PACKAGE

The SIGPLOT plotting package developed at Sigma Research Corporation is a versatile tool for producing different kinds of two-dimensional plots, such as scatter plots, graphs, box plots (sometimes called residual or whisker plots), or error bar plots. The user can specify many parameters including the number of frames per page, the aspect ratio of the frame, and the mapping of the coordinates. The graphics library routines used by SIGPLOT, together with the screen and printer drivers (described later) were originally developed by Dr. Arlindo daSilva of the University of Wisconsin at Milwaukee.

SIGPLOT requires two input files: (1) the template file that contains the control parameters which influence the appearance of the plots, and (2) the input data file that contains the data to be plotted. Tables A-1 and A-2 describe the formats of the template file and the input data file, respectively. Examples of the template file are shown in Figures A-1 through A-4. Examples of the input data file are shown in Figures A-5 through A-8.

SIGPLOT creates a Tektronix picture file that can be viewed directly on any kind of the PC graphics environments (for example, Hercules, CGA, EGA, and VGA) using the screen driver, TEKPC. Hard copy output can also be generated from the Tektronix picture file with a printer driver. There are three printer drives, TEKEPS, TEKELQ, and PS, that are currently available. The first two drivers are used to drive an EPSON-compatible dot matrix printer, with TEKEPS for low resolution and TEKELQ for high resolution. The PS program is used to drive a PostScript printer, such as Apple LaserWriter, NEC LC-890, or TI MicroLaser PS35. It is recommended that the user have access to a PostScript printer to obtain the best results in the shortest time.

SIGPLOT requires about 200KB of memory. The other screen and printer drivers require less than 100KB of memory, except for TEKELQ, where 450 KB of memory is required due to the high resolution and the use of the bitmap approach in the driver program. The SIGPLOT plotting package and the graphics library routines were written in FORTRAN. The screen and printer drivers were written in C.

TABLE A-1. THE FORMAT OF THE TEMPLATE FILE OF SIGPLOT. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL.

The global control parameters are specified in the first section of the template file, lines 1 through 16.

LINE NO.	FORMAT	DESCRIPTION
1-3		Reserved for comments
4	FF/C	Name of the input data file, currently not used
5	FF/C	Name of the output Tektronix picture file, currently not used
6	FF/I	Flag for the frame aspect ratio, 1-5, 1: x:y = 1:1 2: x:y = 1:2 3: x:y = 2:1 4: x:y = 1:3 5: x:y = 3:1
7	FF/I	Number of frames per page, 1-4
8-9	A80	Title for the page (no title will be drawn if "0" appears as the first character of the line)
10	FF/C	Flag (PAXIS) for the axis along which the first column, representing the independent variable, of the data in the input data file (see Table A-2) will be plotted (x or y). PAXIS must = x if IPATTN (described below) = 4, and PAXIS must = y if IPATTN = 6 or 7
11	FF/I	Flag (LTYP) for mapping, 1-4, 1: linear in x, linear in y 2: linear in x, logarithmic in y 3: logarithmic in x, linear in y 4: logarithmic in x, logarithmic in y LTYP must = 4 if IPATTN (described below) = 7

TABLE A-1. THE FORMAT OF THE TEMPLATE FILE OF SIGPLOT. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL.

LINE NO.	FORMAT	DESCRIPTION
12	FF/I	Flag (IPATTN) for plot pattern, 1-7, 1: scatter plot 2: line graph 3: scatter plot except line graph for the last variable 4: box plot 5: error bar plot 6: same as 5 but with extra labelling 7: same as 5 but with extra labelling See more discussion of IPATTN in text
13	FF/I	Flag for background, 0 or 2, 0: no background 2: gridded background
14	FF/C	Flag for system time, y or n, if y: system time will be printed out on the upper right corner of each page
15	5A1	Five point patterns for the scatter plot
16	FF/I	Flag (IEXTRA) for the plotting of extra lines, 1: x=0 will be plotted 2: y=0 will be plotted 3: x=0 and y=0 will be plotted 4: x=1 will be plotted 5: y=1 will be plotted 6: x=1 and y=1 will be plotted 7: diagonal line will be plotted 8: y=0.5 and y=2 (factor of two) will be plotted

TABLE A-1. THE FORMAT OF THE TEMPLATE FILE OF SIGPLOT. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL.

LINE NO.	FORMAT	DESCRIPTION
16 (Concluded)	FF/I	9: $x=-0.667$, 0, and 0.667, and $y=4x^2/(4-x^2)$ (see text) will be plotted if IPATTN = 6; $x=0.5$, 1, and 2 and $y=\exp[(\ell n x)^2]$ (see text) will be plotted if IPATTN = 7. else: no extra lines will be plotted. Note that IEXTRA = 9 is effective only if IPATTN = 6 or 7

The next section of the template file (lines 17 through 29) contains the parameters that are applicable to a frame. This section can be repeated if there are multiple frames to be plotted in a print job. However, the user can prepare just one such section if the same information is to be used repeatedly by all frames.

LINE NO.	FORMAT	DESCRIPTION
17-19		Reserved for comments
20	FF/R	Constants, a and b, for the linear transformation of the independent variable, where $x_{new} = a*x_{old} + b$, a=1 and b=0 means no transformation is needed
21	FF/R	Constants, a and b, for the linear transformation of the first dependent variable, where $y_{1,new} = a*y_{1,old} + b$, a=1 and b=0 means no transformation is needed

TABLE A-1. THE FORMAT OF THE TEMPLATE FILE OF SIGPLOT. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL (CONCLUDED).

LINE NO.	FORMAT	DESCRIPTION
22	FF/R	Same as above, but for the second dependent variable
23	FF/R	Same as above, but for the third dependent variable
24	FF/R	Same as above, but for the fourth dependent variable
25	FF/R	Same as above, but for the fifth dependent variable. Note that lines 22 through 25 cannot be omitted even if only one group of data were to be plotted
26	FF/R	xmin, xmax, and dx of the x-axis
27	FF/R	ymin, ymax, and dy of the y-axis
28	FF/C	Format specifier for the numerical labels of the x-axis. If "!" appears as the first character of the line, the appropriate format will be determined internally by the program; otherwise, the user should supply a simple FORTRAN I-, F-, or E-format specifier, enclosed in parentheses, for example (I5), (F6.3), and (E8.1) are accepted, but (3I5), (I5,f6.3), (1P,E8.1), and (G9.1) are not accepted
29	FF/C	Format specifier for the numerical labels of the y-axis

TABLE A-2. THE FORMAT OF THE INPUT DATA FILE OF SIGPLOT. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL.

LINE NO.	FORMAT	DESCRIPTION
1	A40	Title for the frame (no title will be drawn if "0" appears as the first character of the line)
2	A40	Label for the x-axis (no label will be drawn if "0" appears as the first character of the line)
3	A40	Label for the y-axis (no label will be drawn if "0" appears as the first character of the line)
4	FF/I	Two integers specifying the number of points (NPTS) and the number of groups of data (MANY) to be plotted. MANY cannot be > 5 for IPATTN = 1, 2, 3, and 5, and MANY must be = 1 for IPATTN = 4 and 6. NPTS cannot be > 700 for IPATTN = 1, 2, and 3. NPTS cannot be > 50 for IPATTN = 4, 5, 6, and 7 (see text).
Next NPTS lines: For IPATTN = 1, 2, and 3,		
	FF/R	There are 1+MANY real numbers in each line. The first number represents the independent variable, which can be plotted either along the x- or the y-axis depending the value of PAXIS (see Table A-1). The next MANY numbers represent the dependent variables. For example, if three curves (MANY=3), $f_1(x)$, $f_2(x)$, and $f_3(x)$ were to be plotted, then each line here should contain four real numbers, x_i, $f_{1,i}$ $f_{2,i}$, and $f_{3,i}$, where i=1, NPTS. If PAXIS = "x", the x will be plotted along the abscissa, and f_1, f_2, and f_3 will be plotted along the ordinate; vice versa PAXIS = "y".

TABLE A-2. THE FORMAT OF THE INPUT DATA FILE OF SIGPLOT. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL.

LINE NO.	FORMAT	DESCRIPTION
For IPATTN = 4,		
	FF/R, I	There are six real numbers and one integer in each line. The first real number represents the independent variable. The next five real numbers represent the values of the dependent variable at the 2nd, 16th, 50th, 84th, and 98th percentiles, respectively. Note that the value of the independent variable listed here frequently represents a range of the independent variable; for example, a wind speed of 7 m/s actually represents wind speeds in the range of 6 to 8 m/s. The integer represents the number of data points based on which the distribution of the dependent variable is derived. No box will be plotted if the number of data points is less than five since not enough information is available to define a distribution.
For IPATTN = 5,		
	FF/R	There are 1+3*MANY real numbers in each line. The first number represents the independent variable. The remaining numbers for the dependent variables are in MANY groups of three numbers. The three numbers, which must be in order, represent the distribution of a dependent variable. This distribution can be 1) $\mu-\sigma$, μ, and $\mu+\sigma$, where μ is the mean, and σ is the standard deviation, or 2) lower c.l., nominal value, and upper c.l., where c.l. is the confidence limit.

TABLE A-2. THE FORMAT OF THE INPUT DATA FILE OF SIGPLOT. THE FOLLOWING KEY LETTERS ARE USED IN THE FORMAT COLUMN - FF: FREE FORMAT, C: CHARACTER, I: INTEGER, AND R: REAL (CONCLUDED).

LINE NO.	FORMAT	DESCRIPTION
For IPATTN = 6 and 7,		
	FF/R,C	There are four real numbers and one character constant (no more than 17 characters long) in each line. The definition of the first four real numbers is identical to that when IPATTN = 5, except now MANY must = 1. The character constant, enclosed in apostrophes, is used to label each data point.

The above 4+NPTS lines provide enough information to plot a frame. Additional data, similar in structure, can be appended here if the plotting of more than one frames in a print job is desired.

```
!---------------------------------------------------------------------
!      Main switches for plotting.
!---------0---------0---------0---------0-----------------------------
urrs.1    Name of input data file.
tekl.pic  Name of output tektronix file.
1         Aspect ratio (integer, 1 - 5).
1         Number of plots per page (integer, 1 - 4).
demo of ipattn=2
0
x         Which axis serves as independent variable (x or y).
1         Flag indicating log or linear mapping (1 - 4).
2         Pattern.
0         Background specification.
y         Print out system time on the upper right hand corner (y or n).
.+o#$     Patterns of scatter plots (5a1)
0   Extra line,1:x=0,2:y=0,3:x,y=0,4:x=1,5:y=1,6:x,y=1,7:diag,8:y=fac. 2.,9:fb-nmse, else:nothing.
!---------------------------------------------------------------------
!      Parameters for plot 1.
!---------------------------------------------------------------------
1. 0.     ascale, bscale for the independent variable axis.
1. 0.     ascale, bscale for curve 1.
1. 0.     ascale, bscale for curve 2.
1. 0.     ascale, bscale for curve 3.
1. 0.     ascale, bscale for curve 4.
1. 0.     ascale, bscale for curve 5.
-6.28319 6.28319 3.141595  xmin, xmax, and dx for the x axis.
-1.2 1.2 0.3 ymin, ymax, and dy for the y axis.
(f5.2)          format for x label
(f4.1)        format for y label
```

Figure A-1. An Example of the Template File of SIGPLOT. Refer to Figure A-10 for the Results.

```
!------------------------------------------------------------------
!       Main switches for plotting.
!---------0---------0---------0---------0--------------------------
urrs.1    Name of input data file.
tekl.pic  Name of output tektronix file.
1         Aspect ratio (integer, 1 - 5).
4         Number of plots per page (integer, 1 - 4).
demo of ipattn=4
0
x         Which axis serves as independent variable (x or y).
2         Flag indicating log or linear mapping (1 - 4).
4         Pattern.
0         Background specification.
n         Print out system time on the upper right hand corner (y or n).
m.+o#     Patterns of scatter plots (5a1)
8   Extra line,1:x=0,2:y=0,3:x,y=0,4:x=1,5:y=1,6:x,y=1,7:diag,8:y=fac. 2.,9:fb-nmse, else:nothing.
!------------------------------------------------------------------
!       Parameters for plot 1.
!------------------------------------------------------------------
1. 0.     ascale, bscale for the independent variable axis.
1. 0.     ascale, bscale for curve 1.
1. 0.     ascale, bscale for curve 2.
1. 0.     ascale, bscale for curve 3.
1. 0.     ascale, bscale for curve 4.
1. 0.     ascale, bscale for curve 5.
0 10 2 xmin, xmax, and dx for the x axis.
0.01 100. 10.  ymin, ymax, and dy for the y axis.
(i2)
(f6.2)
!------------------------------------------------------------------
!       Parameters for plot 2. (stability class)
!------------------------------------------------------------------
1. 0.     ascale, bscale for the independent variable axis.
1. 0.     ascale, bscale for curve 1.
1. 0.     ascale, bscale for curve 2.
1. 0.     ascale, bscale for curve 3.
1. 0.     ascale, bscale for curve 4.
1. 0.     ascale, bscale for curve 5.
-2. 2. 0. xmin, xmax, and dx for the x axis.
0.01 100. 10.  ymin, ymax, and dy for the y axis.
(i2)
(f6.2)
!------------------------------------------------------------------
!       Parameters for plot 3. (mixing height)
!------------------------------------------------------------------
1. 0.     ascale, bscale for the independent variable axis.
1. 0.     ascale, bscale for curve 1.
1. 0.     ascale, bscale for curve 2.
1. 0.     ascale, bscale for curve 3.
1. 0.     ascale, bscale for curve 4.
1. 0.     ascale, bscale for curve 5.
0. 30. 5. xmin, xmax, and dx for the x axis.
0.01 100. 10.  ymin, ymax, and dy for the y axis.
(i2)
(f6.2)
!------------------------------------------------------------------
!       Parameters for plot 4. (hour of day)
!------------------------------------------------------------------
1. 0.     ascale, bscale for the independent variable axis.
1. 0.     ascale, bscale for curve 1.
1. 0.     ascale, bscale for curve 2.
1. 0.     ascale, bscale for curve 3.
1. 0.     ascale, bscale for curve 4.
1. 0.     ascale, bscale for curve 5.
0. 24. 4. xmin, xmax, and dx for the x axis.
0.01 100. 10.  ymin, ymax, and dy for the y axis.
(i2)
(f6.2)
```

Figure A-2. An Example of the Template File of SIGPLOT. Refer to Figure A-12 for the Results.

```
!-----------------------------------------------------------------------
!      Main switches for plotting.
!--------0---------0---------0---------0--------------------------------
urrs.1    Name of input data file.
tekl.pic  Name of output tektronix file.
1         Aspect ratio (integer, 1 - 5).
1         Number of plots per page (integer, 1 - 4).
demo of ipattn=5
0
x         Which axis serves as independent variable (x or y).
3         Flag indicating log or linear mapping (1 - 4).
5         Pattern.
0         Background specification. (0 or 2)
n         Print out system time on the upper right hand corner (y or n).
+o.#$     Patterns of scatter plots (5a1)
0   Extra line,1:x=0,2:y=0,3:x,y=0,4:x=1,5:y=1,6:x,y=1,7:diag,8:y=fac. 2.,9:fb-nmse, else:nothing.
!-----------------------------------------------------------------------
!      Parameters for plot 1.
!-----------------------------------------------------------------------
1. 0.     ascale, bscale for the independent variable axis.
1. 0.     ascale, bscale for curve 1.
1. 0.     ascale, bscale for curve 2.
1. 0.     ascale, bscale for curve 3.
1. 0.     ascale, bscale for curve 4.
1. 0.     ascale, bscale for curve 5.
200. 20000. 10. xmin, xmax, and dx for the x axis.
-1.5 1.5 0.5  ymin, ymax, and dy for the y axis.
(i5)
(f4.1)
```

Figure A-3. An Example of the Template File of SIGPLOT. Refer to Figure A-13 for the Results.

```
!------------------------------------------------------------------------
!      Main switches for plotting.
!--------0---------0---------0---------0--------------------------------
urrs.1    Name of input data file.
tekl.pic  Name of output tektronix file.
1         Aspect ratio (integer, 1 - 5).
1         Number of plots per page (integer, 1 - 4).
demo of ipattn=6
0
y         Which axis serves as independent variable (x or y).
1         Flag indicating log or linear mapping (1 - 4).
6         Pattern.
0         Background specification. (0 or 2)
n         Print out system time on the upper right hand corner (y or n).
+o.#$     Patterns of scatter plots (5al)
9   Extra line,1:x=0,2:y=0,3:x,y=0,4:x=1,5:y=1,6:x,y=1,7:diag,8:y=fac. 2.,9:fb-nmse, else:nothing.
!------------------------------------------------------------------------
!      Parameters for plot 1.
!------------------------------------------------------------------------
1. 0.     ascale, bscale for the independent variable axis.
1. 0.     ascale, bscale for curve 1.
1. 0.     ascale, bscale for curve 2.
1. 0.     ascale, bscale for curve 3.
1. 0.     ascale, bscale for curve 4.
1. 0.     ascale, bscale for curve 5.
-2. 2. 0.5  xmin, xmax, and dx for the x axis.
0. 15. 2.5  ymin, ymax, and dy for the y axis.
(f5.1)
(f4.1)
```

Figure A-4. An Example of the Template File of SIGPLOT. Refer to Figure A-14 for the Results.

```
0
x
y
         50 5
   -6.03186      0.248690     -0.368124     -0.844328     -0.998027     -0.770514
   -5.78053      0.481754     -0.125333     -0.684547     -0.982287     -0.904827
   -5.52920      0.684547      0.125333     -0.481753     -0.904827     -0.982287
   -5.27788      0.844328      0.368125     -0.248690     -0.770513     -0.998027
   -5.02655      0.951057      0.587786      0.397359E-06 -0.587785     -0.951056
   -4.77522      0.998027      0.770513      0.248690     -0.368125     -0.844328
   -4.52389      0.982287      0.904827      0.481754     -0.125333     -0.684547
   -4.27257      0.904827      0.982287      0.684547      0.125333     -0.481753
   -4.02124      0.770513      0.998027      0.844328      0.368125     -0.248690
   -3.76991      0.587785      0.951056      0.951057      0.587785      0.254308E-06
   -3.51858      0.368125      0.844328      0.998027      0.770513      0.248690
   -3.26726      0.125333      0.684547      0.982287      0.904827      0.481754
   -3.01593     -0.125333      0.481753      0.904827      0.982287      0.684547
   -2.76460     -0.368125      0.248690      0.770513      0.998027      0.844328
   -2.51327     -0.587785      0.397391E-07  0.587785      0.951056      0.951057
   -2.26195     -0.770513     -0.248690      0.368124      0.844328      0.998027
   -2.01062     -0.904827     -0.481754      0.125333      0.684547      0.982287
   -1.75929     -0.982287     -0.684547     -0.125334      0.481753      0.904827
   -1.50796     -0.998027     -0.844328     -0.368124      0.248690      0.770513
   -1.25664     -0.951057     -0.951057     -0.587785     -0.556284E-07  0.587785
   -1.00531     -0.844328     -0.998027     -0.770513     -0.248690      0.368124
   -0.753983    -0.684547     -0.982287     -0.904827     -0.481753      0.125333
   -0.502655    -0.481754     -0.904827     -0.982287     -0.684547     -0.125333
   -0.251328    -0.248690     -0.770513     -0.998027     -0.844328     -0.368125
    0.000000     0.000000     -0.587785     -0.951056     -0.951057     -0.587785
    0.251328     0.248690     -0.368124     -0.844328     -0.998027     -0.770513
    0.502655     0.481753     -0.125333     -0.684547     -0.982287     -0.904827
    0.753982     0.684547      0.125333     -0.481754     -0.904827     -0.982287
    1.00531      0.844328      0.368125     -0.248690     -0.770513     -0.998027
    1.25664      0.951056      0.587785     -0.381470E-06 -0.587786     -0.951057
    1.50796      0.998027      0.770513      0.248690     -0.368125     -0.844328
    1.75929      0.982287      0.904827      0.481754     -0.125333     -0.684547
    2.01062      0.904827      0.982287      0.684547      0.125333     -0.481754
    2.26195      0.770513      0.998027      0.844328      0.368125     -0.248690
    2.51327      0.587785      0.951056      0.951057      0.587785      0.190735E-06
    2.76460      0.368124      0.844328      0.998027      0.770513      0.248690
    3.01593      0.125334      0.684548      0.982287      0.904827      0.481753
    3.26726     -0.125333      0.481754      0.904827      0.982287      0.684547
    3.51858     -0.368124      0.248690      0.770514      0.998027      0.844328
    3.76991     -0.587785      0.341731E-06  0.587785      0.951057      0.951056
    4.02124     -0.770513     -0.248690      0.368125      0.844328      0.998027
    4.27257     -0.904827     -0.481754      0.125333      0.684547      0.982287
    4.52389     -0.982287     -0.684547     -0.125333      0.481754      0.904827
    4.77522     -0.998027     -0.844327     -0.368124      0.248691      0.770514
    5.02655     -0.951057     -0.951056     -0.587785      0.723200E-06  0.587786
    5.27787     -0.844328     -0.998027     -0.770513     -0.248689      0.368125
    5.52920     -0.684548     -0.982287     -0.904827     -0.481753      0.125334
    5.78053     -0.481754     -0.904827     -0.982287     -0.684547     -0.125333
    6.03186     -0.248690     -0.770513     -0.998027     -0.844328     -0.368124
    6.28319     -0.301992E-06 -0.587785     -0.951057     -0.951056     -0.587785
```

Figure A-5. An Example of the Input Data File of SIGPLOT. Refer to Figure A-10 for the Results.

```
0
Wind Speed (m/s)
pred / obs
     4     1
     2.0000     0.4975     0.6268     1.0008     1.6317     1.7629     9
     3.0000     0.5616     0.7442     1.1301     1.7035     6.0669    21
     4.0000     0.0100     0.0100     0.7461     1.6532     2.3788    17
     5.0000     0.1280     0.6695     1.1975     1.5072     1.8114    25
0
Stability index
pred / obs
     3     1
    -1.0000     0.7775     1.0573     1.1854     1.4475     1.6532    15
     0.0000     0.3899     0.7461     0.9782     1.6317     3.0062    40
     1.0000     0.0100     0.0638     0.7442     1.5230     6.0669    24
0
Mixing height *100 (m)
pred / obs
     5     1
     0.9950     0.0100     0.0100     0.6630     1.5081     1.7379     9
     4.0000     0.6034     0.8437     1.3918     1.9281     3.0062     9
    12.5000     0.3899     0.5616     0.9782     1.4252     6.0669    22
    17.5000     0.7775     0.8599     1.2784     1.5724     2.3788    21
    25.0000     0.0100     0.4864     0.7847     1.2369     1.5230    16
0
Hour ending
pred / obs
     6     1
     1.9950     0.0100     0.0100     0.8539     1.3918     3.0062     8
     6.0000     0.4975     0.5616     0.8437     1.7379     1.9281    12
    10.0000     0.5906     0.6061     0.9806     1.1854     1.3115    12
    14.0000     0.6695     0.7775     1.1975     1.5072     1.6532    15
    18.0000     0.3899     0.4864     0.9782     1.8114     2.3788    17
    22.0050     0.0100     0.1280     1.2369     1.7035     6.0669    15
```

Figure A-6. An Example of the Input Data File of SIGPLOT. Refer to Figure A-12 for the Results.

```
all periods
n-s distance (m)
var(dws) / median 1-min var(ws) /2
6  3
     312.5 -0.029  0.002  0.034 -0.027  0.043  0.112  0.166  0.275  0.383
     625.0 -0.009  0.005  0.019  0.011  0.048  0.085  0.209  0.361  0.513
    1250.0 -0.035  0.012  0.059 -0.007  0.070  0.148  0.330  0.480  0.630
    2500.0 -0.155  0.015  0.185 -0.126  0.100  0.325  0.359  0.565  0.771
    5000.0 -0.033  0.170  0.373  0.013  0.323  0.633  0.440  0.751  1.062
   10000.0 -0.417  0.061  0.539 -0.137  0.369  0.876  0.406  0.873  1.339
```

Figure A-7. An Example of the Input Data File of SIGPLOT. Refer to Figure A-13 for the Results.

```
Thorney Island, instantaneous
FB (with 95-percent c.i.)
NMSE
        11 1
   10.4487       -1.61583       -1.53316       -1.44948      'AFTOX'
   0.216873      -0.264945      -0.161629      -0.522500E-01 'AIRTOX'
   0.289691      -0.410913      -0.315094      -0.221415     'BM'
   0.456305       0.303394       0.452075       0.599277     'CHARM'
   1.35801       -0.812071      -0.664101      -0.510295     'DEGADIS'
   1.37463       -0.937623      -0.841639      -0.740987     'FOCUS'
   0.300757       0.322819       0.427639       0.528877     'GASTAR'
   2.31389       -1.16484       -1.07991       -0.996074     'INPUFF'
   0.431075      -0.563486      -0.479006      -0.398916     'PHAST'
   0.283326       0.393132       0.467431       0.540532     'SLAB'
   2.06659       -1.03175       -0.911701      -0.777825     'TRACE'
```

Figure A-8. An Example of the Input Data File of SIGPLOT. Refer to Figure A-14 for the Results.

As one can see from Table A-1, SIGPLOT is capable of creating the following kinds of plots:

IPATTN = 1:	scatter plot (for example, Figure A-9)
IPATTN = 2:	line graph (for example, Figure A-10)
IPATTN = 3:	scatter plot except line graph for the last variable (for example, Figure A-11)
IPATTN = 4:	box plot (for example, Figure A-12)
IPATTN = 5:	error bar plot (for example, Figure A-13)
IPATTN = 6 and 7:	same as IPATTN = 5 but with extra labelling (for example, Figures A-14 and A-15)

The usage of each option is described below.

For IPATTN = 1, groups of data are represented by different dot patterns that are defined in the template file (see Table A-1). At most, five groups of data (MANY = 5) can be plotted, with a maximum of 700 points for each group.

IPATTN = 2 is similar to IPATTN = 1 except that points are now connected. The following line patterns are used to represent different curves: solid, short-dashed, long-dashed, dot-dashed, and dotted. At most, five curves (MANY = 5) can be plotted, with a maximum of 700 points for each group. No user customization of the line patterns is allowed. It is important that the data points in the input file are sorted according to the independent variable.

IPATTN = 3, a combination of IPATTN = 1 and 2, is useful when the user wants to see how well a theoretical curve fits the observed data. Although the order of the data points does not matter for a scatter plot, in this case it is important that the data points in the input file are sorted according to the independent variable. At most, five groups of data (MANY = 5) can be plotted, with a maximum of 700 points for each group.

The IPATTN = 4 option is an alternative to the scatter plot when the number of data points is large. In preparing the input data file for SIGPLOT,

DEMO OF IPATTN=1

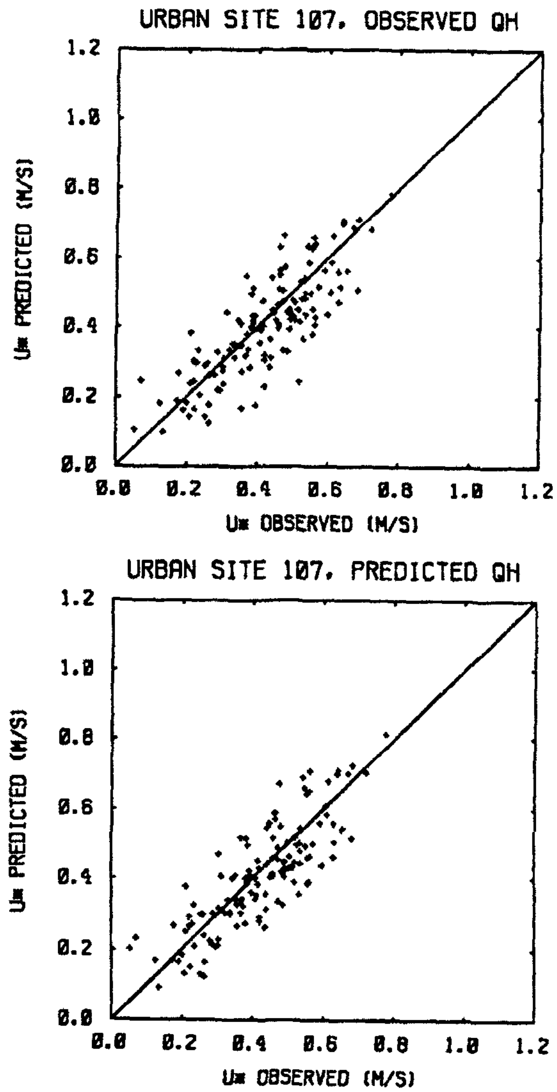


Figure A-9. A Sample Scatter Plot (IPATTN = 1) Generated by SIGPLOT.

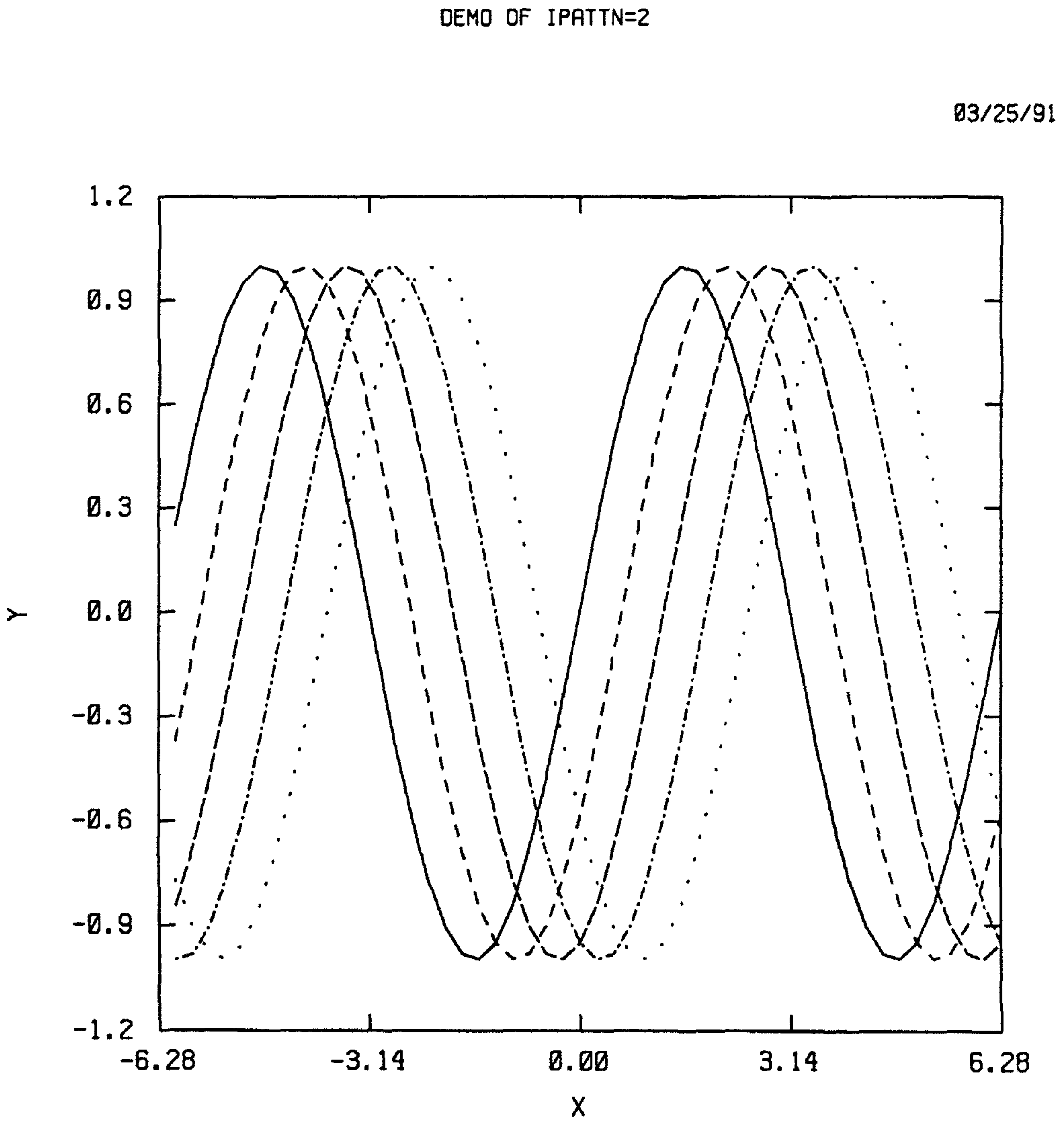


Figure A-10. A Sample Line Graph (IPATTN = 2) Generated by SIGPLOT. Refer to Figures A-1 and A-5 for the Template and Data Files used for this Figure.

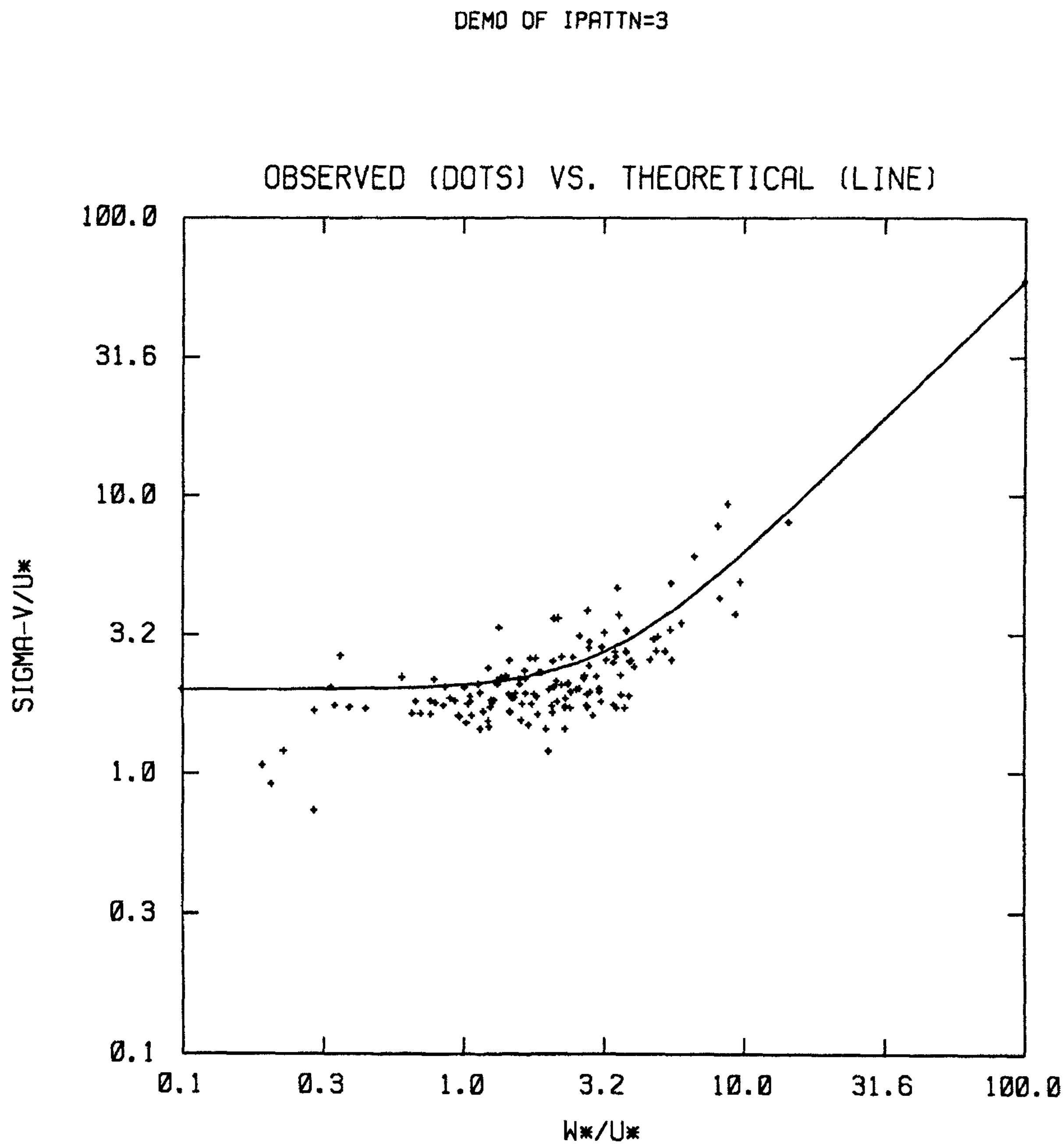


Figure A-11. A Sample Scatter Plot and Line Graph (IPATTN = 3) Generated by SIGPLOT.

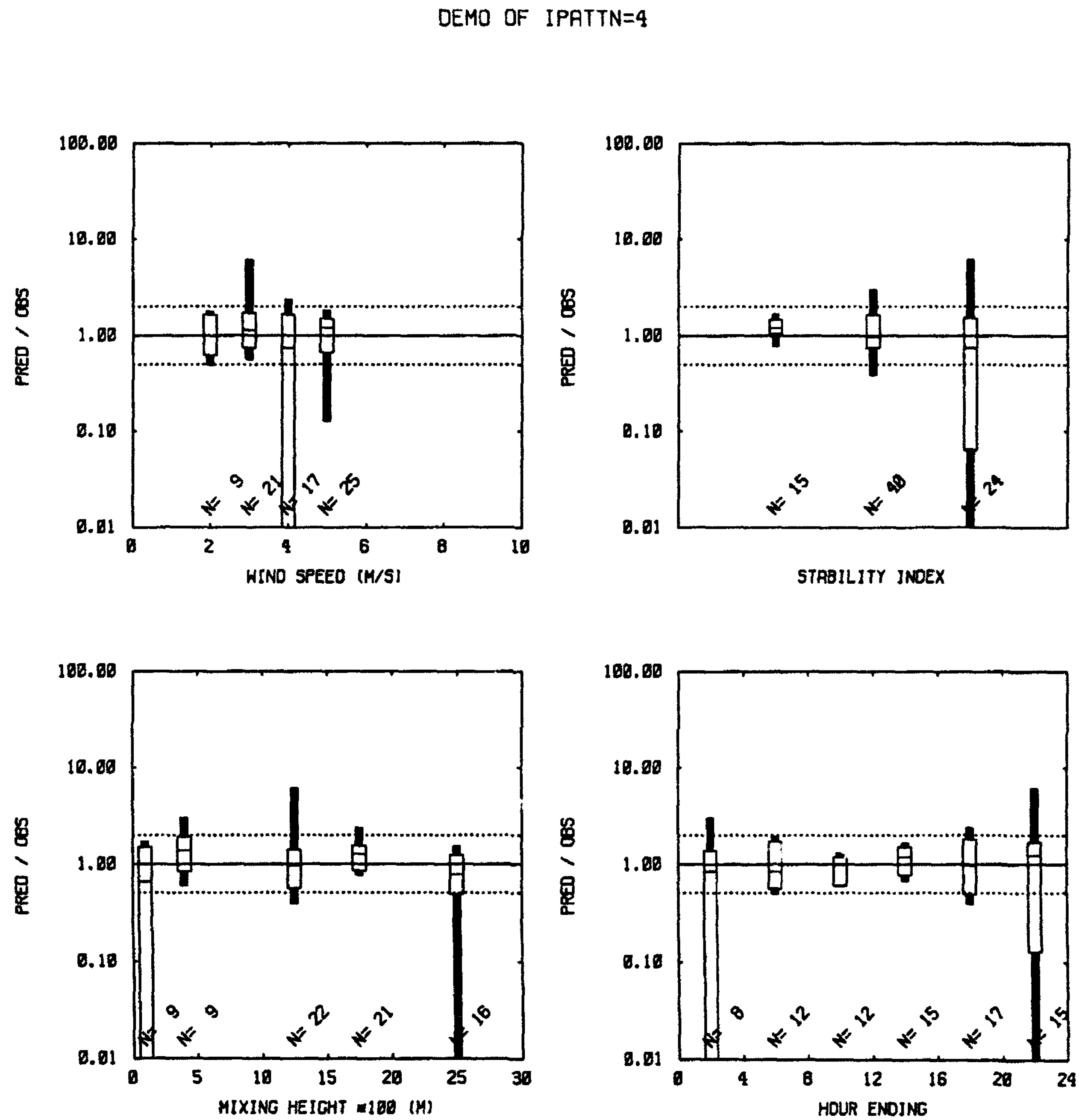


Figure A-12. A Sample Box Plot (IPATTN = 4) Generated by SIGPLOT. Refer to Figures A-2 and A-6 for the Template and Data Files used for this Figure.

DEMO OF IPATTN=5

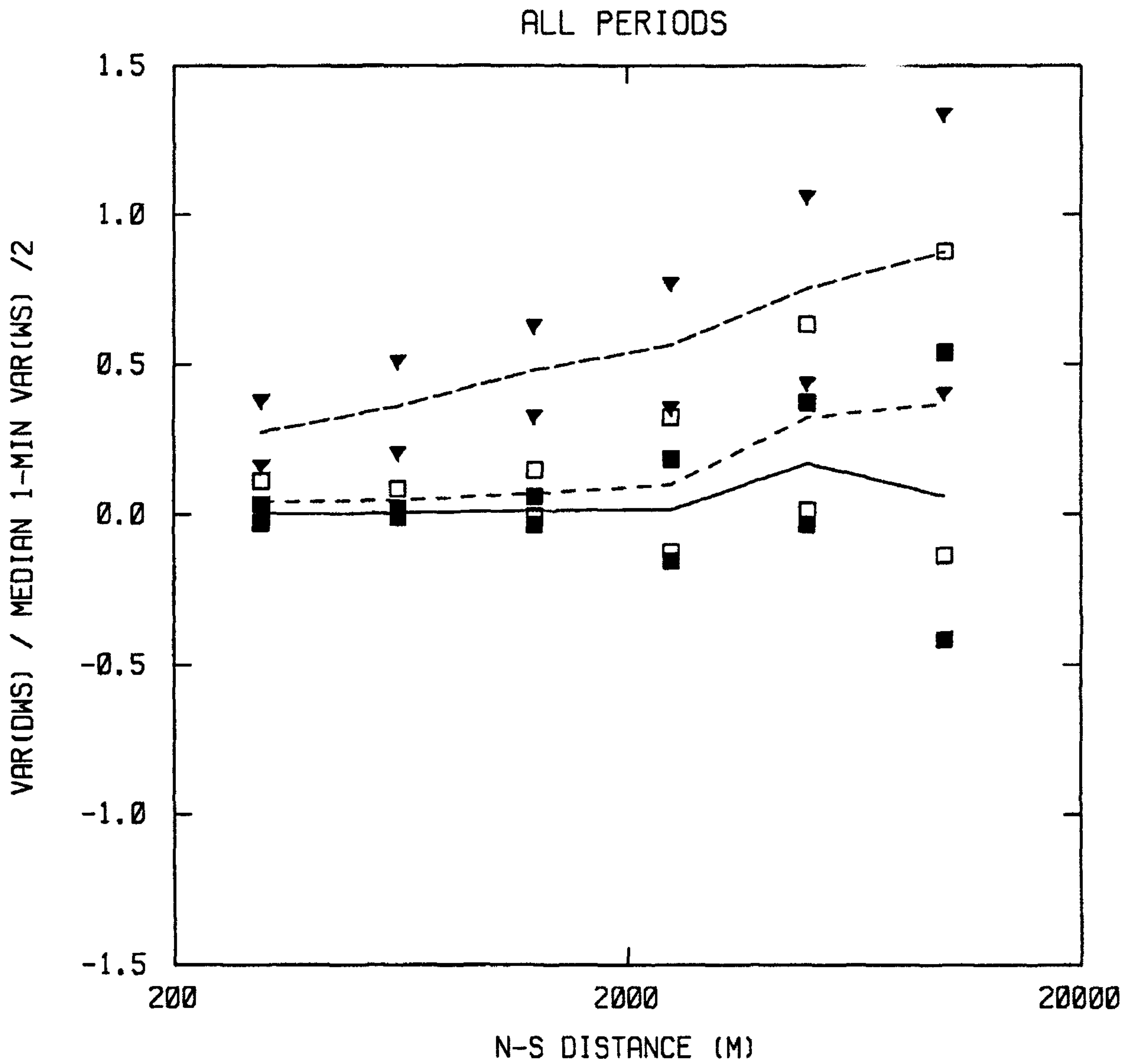


Figure A-13. A Sample Error Bar Plot (IPATTN = 5) Generated by SIGPLOT. Refer to Figures A-3 and A-7 for the Template and Data Files used for this Figure.

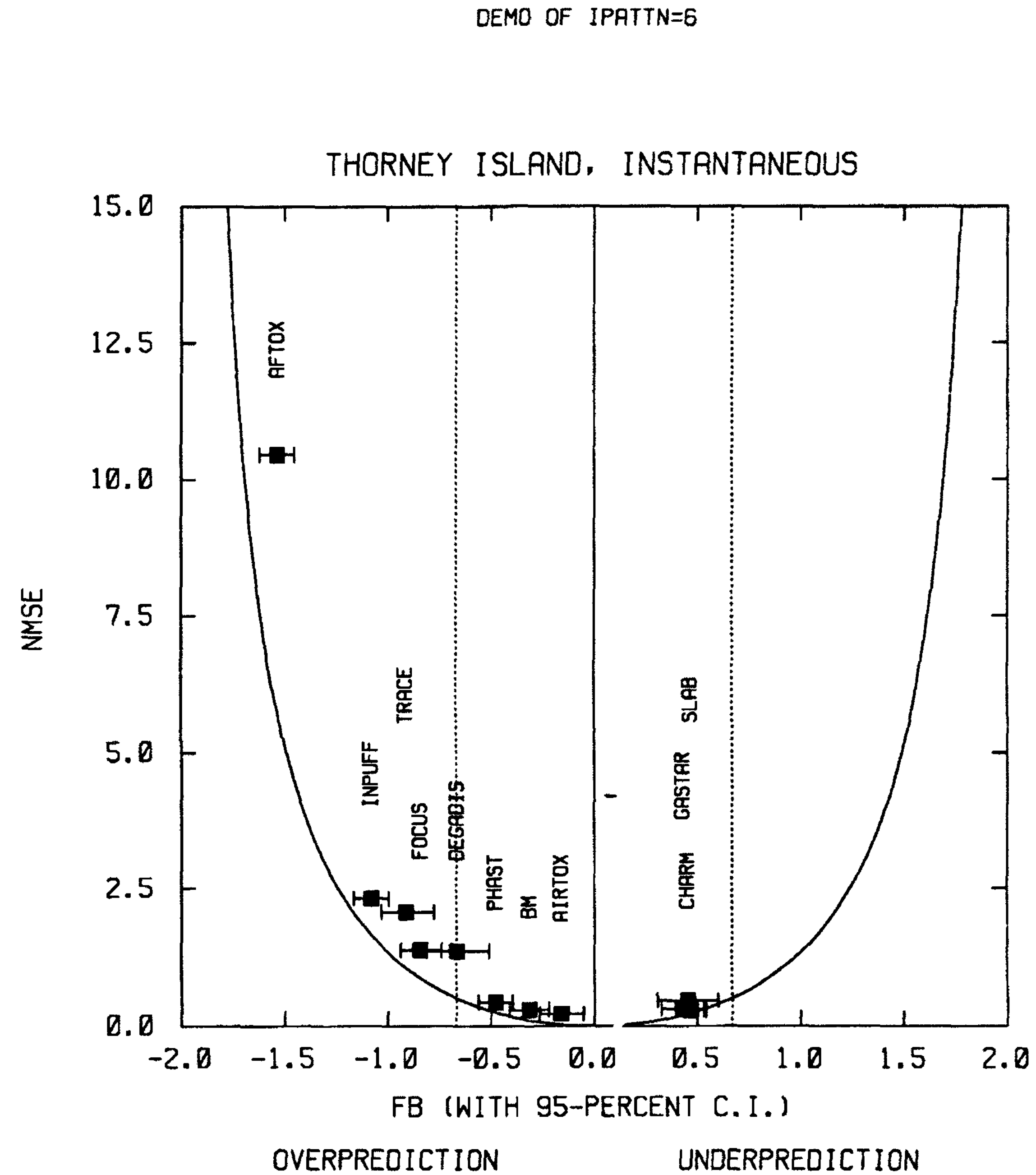


Figure A-14. A Sample Error Bar Plot with Labelling (IPATTN = 6) Generated by SIGPLOT. Refer to Figure A-4 and A-8 for the Template and Data Files used for this Figure.

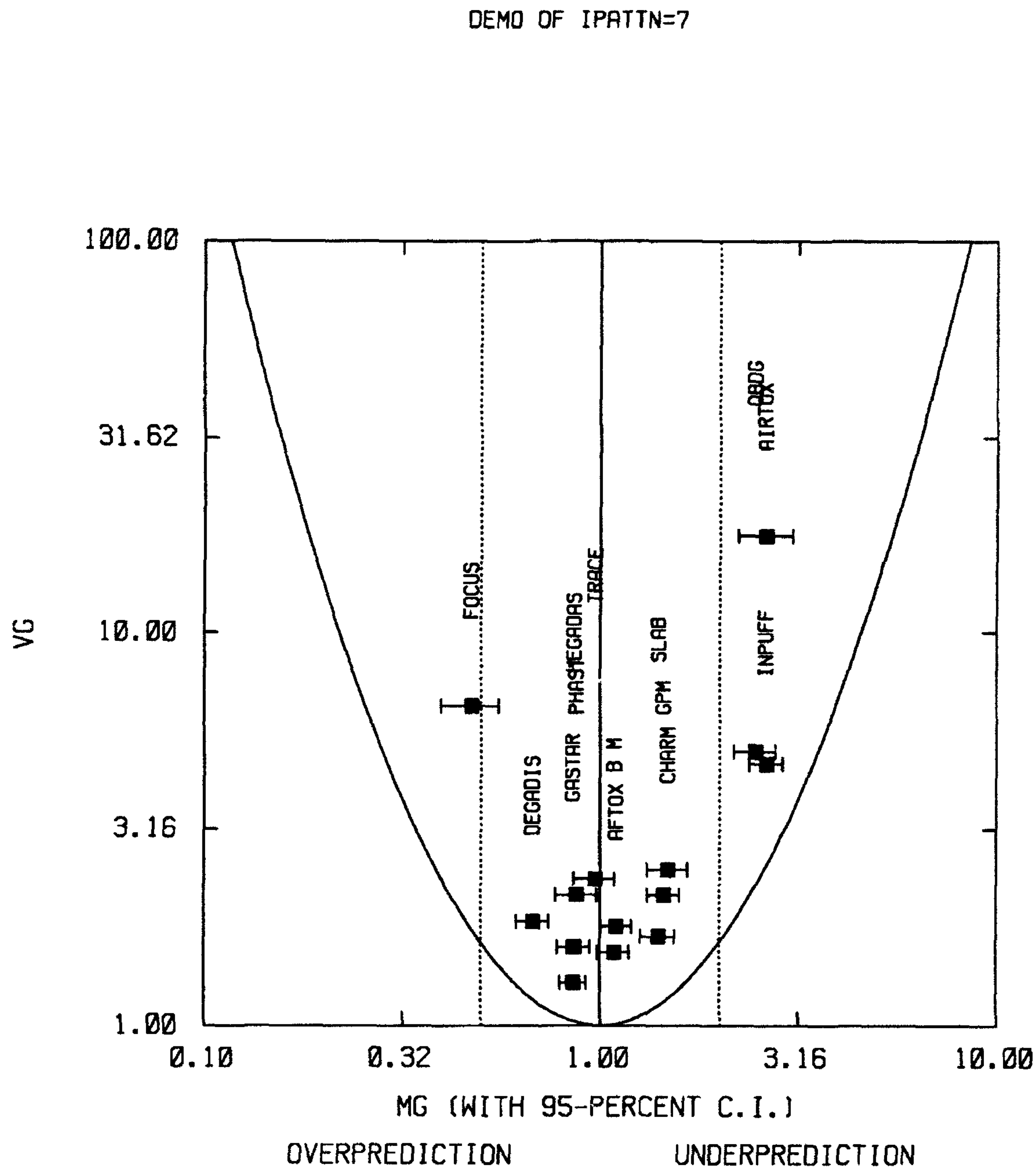


Figure A-15. A Sample Error Bar Plot with Labeling (IPATTN = 7) Generated by SIGPLOT.

the user first defines certain ranges of the independent variable to be used for grouping the dependent variable. The distribution of the dependent variables within each group is then determined and represented by five significant points in the cumulative distribution function (cdf). These five values could be the 2nd, 16th, 50th, 84th, and 98th percentiles of the cdf, or the mean and mean ± one and two standard deviations. SIGPLOT then uses a box pattern to represent the distribution of the dependent variable within each grouping or range of the independent variable. Only one set of data (that is, MANY = 1, even though five points are needed to define a box) is accepted for this option, with a maximum of 50 boxes.

IPATTN = 5 is similar to IPATTN = 4 except that three values (vs. five) are needed to define an error bar (vs. a box). These three values can be the mean and mean ± one standard deviation of a dependent variable, or the nominal value of a dependent variable and its 95 percent confidence limits. At most, five groups (MANY = 5) of data can be plotted, with a maximum of 50 error bars for each group. The following error bar patterns are used: filled square, empty square, filled triangle, empty triangle, and cross.

IPATTN = 6 is similar to IPATTN = 5 except that the user can label each data point. Because of the additional information to be plotted, only one group of data (MANY = 1) is accepted, with a maximum of 50 error bars. This option is designed primarily to plot the FB (fractional bias), together with its confidence limits, against the NMSE (normalized mean square error), where

$$FB = \frac{(\bar{C}_o - \bar{C}_p)}{(0.5(\bar{C}_o + \bar{C}_p))} \tag{A-1}$$

$$NMSE = \frac{\overline{(C_o - C_p)^2}}{\overline{C_o C_p}} \tag{A-2}$$

If IEXTRA = 9 (see Table A-1), SIGPLOT will plot the additional x = -0.667, 0, and 0.667 lines, representing the factor of two and zero FB lines, together with the $y = 4x^2/(4-x^2)$ line, representing the "minimum" NMSE (due only to the mean bias) as a function of FB.

IPATTN = 7 is identical to IPATTN = 6, except that it is designed primarily to plot the MG (geometric mean bias), together with its confidence limits, against the VG (geometric mean variance), where

$$MG = \exp(\overline{\ell n C_o} - \overline{\ell n C_p}) \quad \text{(A-3)}$$

$$VG = \exp[\overline{(\ell n C_o - \ell n C_p)^2}] \quad \text{(A-4)}$$

If IEXTRA = 9 (See Table A-1), SIGPLOT will plot the additional x = 0.5, 1, and 2 lines, representing the factor of two and zero FB lines, together with the $y = \exp[(\ell n x)^2]$ line, representing the "minimum" VG (due only to the mean bias) as a function of MG. Note that it is always ℓn(MG) and ℓn(VG) (that is, LTYP=4, see Table A-1) that are actually plotted.

The instructions for the driver programs, TEKPC, TEKELQ, TEKEPS, and PS, can be obtained by simply executing the programs without providing any arguments, and will not be repeated here.

Finally, an example is given below of the procedures followed to use the graphics package.

Step 1: The user prepares the template file (DEMO.INQ) and the input data file (DEMO.DAT) according to the formats described in Tables A-1 and A-2. The user can create his own template file by editing the sample template file. The input data file is usually generated by some other programs.

Step 2: After the execution of SIGPLOT, a Tektronix picture file (DEMO.PIC) is generated.

Step 3: The user can view the results on screen by typing:
TEKPC DEMO.PIC
if a Hercules graphics card is installed, or
TEKPC DEMO.PIC 16
if an EGA (with a resolution of 640x350 pixels) graphics card is installed.

Step 4: A high resolution hard copy output can be generated on an EPSON-compatible dot matrix printer by typing:
TEKELQ DEMO.PIC.

Step 5: Or if the user has access to a PostScript printer, a PostScript file (DEMO.PS) will be created by typing:
PS DEMO.PIC,
and this file can be printed out by typing:
PRINT DEMO.PS